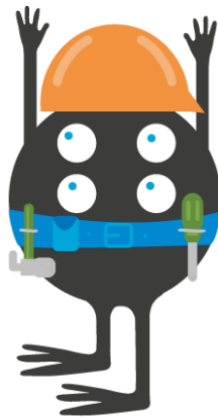




GETTING STARTED: IOS INTEGRATION GUIDE





Technical overview

The Howtank iOS widget is a library that, once included in your app, display the Howtank chat. Before being clicked by the user, it is in a "folded" state, waiting quietly for a user action. Once clicked by the user, it switches to the "expanded" state, exchanging chat data with Howtank servers.

The widget is very lightweight to preserve your application.

Installation

Using CocoaPods

CocoaPods is the dependency manager for Swift and Objective-C Cocoa projects. See more information here: <https://cocoapods.org>

If you already use CocoaPods, just add the following line in your Podfile configuration:

```
pod 'HowtankWidget', '1.7.0'
```

Then run the `pod install` command in your Terminal.

Manual installation

Download the HowtankWidget framework here:

<http://cdn.howtank.com/sdk/ios/HowtankWidget-1.7.0.zip>

Unzip and drag the "Howtank" folder on your "Framework" folder in XCode. Don't forget to check "Copy items if needed" and "Create groups":

- Destination: ☒ Copy items if needed
- Added folders: ☒ Create groups
☐ Create folder references



Quick setup

Widget initialization

To get the HowtankWidget up and running, you just need to add the following lines in your AppDelegate class:

Objective C

```
#import <HowtankWidget/HowtankWidget.h>
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {
    ...

    [[HowtankWidget sharedWidget] initWithHostId:@"HOST_ID" andDelegate:nil];

    return true
}
```

Swift

```
HowtankWidget.sharedWidget().initWithHostId("HOST_ID", andDelegate: nil)
```

Don't forget to import Howtank Framework in your `*-Bridging-Header.h` file:

```
#import <HowtankWidget/HowtankWidget.h>
```

Where:

Name	Description	Mandatory?
HOST_ID	Your identifier (given by Howtank team)	Yes

ViewControllers configuration

You might want to display the widget on some controllers and not on some others, or display the widget everywhere in your app. In any case, you **need** to call the following method within all your viewControllers `viewWillAppear` function:

```
// Howtank Widget specific configuration
HowtankWidget.getInstance().setDiscreet(IS_DISCREET, "PAGE_NAME", "PAGE_URL");
```

Where:

Name	Description	Mandatory?
IS_DISCREET	boolean, if the widget should be “discreet” (hidden by default) (<code>true</code>) on this controller or not (<code>false</code>)	Yes
PAGE_NAME	String, the name of the current controller (ie. “Product page “ or	Yes

Document strictly confidential, Howtank property



	“Product – Apple iPhone”). This information will be read by your community member when taking a chat, so the more precise the better.	
PAGE_URL	String, a URL representation of your current page. Most apps use URLs for deeplinking. Again, this url will be clickable by members, so they can have a precise view of what the user is watching. Examples are : http://www.mywebsite.com/product/1234 or myapp://product/1234	Yes

That’s all! The Howtank Widget should appear on the bottom-right end corner of your application. Please note that it might take a few seconds since we query our servers to decide whether or not the widget should be displayed.





Advanced configuration

Default configuration gets the widget running in a few lines of code. However, you can overload it with the following parameters.

Please note that `initWithHostId` should always be called last.

Example

Init example (Swift):

```
HowtankWidget.sharedWidget()  
    .setVerboseOn(true)  
    .initWithHostId("HOST_ID", andDelegate: nil)
```

Verbose mode

```
setVerboseOn(true|false)
```

Enable more detailed logs when something went wrong. Only sets this to true in debug mode when debugging with the Howtank team.

Display validation when disabling the widget

```
setWidgetDisablingRequiresValidation(true|false)
```

When the user tries to disable the widget, decides whether or not to display a confirmation. Default is `false`.

Adding a delegate

You can register your class as a `HowtankWidgetDelegate` when calling the init method:

```
HowtankWidget.sharedWidget().initWithHostId("HOST_ID", andDelegate: self)
```

The following methods will be called when specific actions occur. They are both optional:

Selected link

```
func howtankWidget(howtankWidget: HowtankWidget, didSelectLink link: String) {  
    // Handle the clicked link  
}
```



When user click on the link in the chat, this method is called so you can handle the clicked link properly. By default, nothing happen when user clicks on a link.

Widget events

```
func howtankWidget(_ howtankWidget: HowtankWidget, event: HowtankWidgetEventType) {  
    // Called when a specific widget event occurs  
}
```

These method is called when a specific event is triggered, usually by the user. The following events may be triggered:

Event name	Description
INITIALIZED	When the widget has been correctly initialized and the chat is active
OPENED	Triggered when the user clicks on the chat bubble
DISABLED	When the chat bubble has been dragged and released over the deletion area
DISPLAYED	When the widget bubble is displayed
HIDDEN	When the widget bubble is hidden

Widget unavailable

```
func howtankWidget(howtankWidget: HowtankWidget, isUnavailable reason: String) {  
    // Called when the widget is unavailable  
}
```

When the widget is unavailable, this method is called with the reason.



Conversion tracker integration

You can track 2 types of goals: generic and purchases.

Generic goal tracking

The following tracker has to be called on controllers where the goals you want to track are achieved:

```
HowtankWidget.sharedWidget().conversion(withType: "GOAL_NAME")
```

Where:

Name	Description	Mandatory?
GOAL_NAME	String, the name of the goal you want to track. You can have several goals (e.g. purchase, registration, subscription, etc.). Our platform will automatically generate a report for each goal tracked.	Yes

Purchase

This specific tracker allows you to provide more information about purchases:

```
let purchaseParameters = HTPurchaseParameters()
purchaseParameters.isNewBuyer = IS_NEW_BUYER
purchaseParameters.purchaseId = "PURCHASE_ID"
purchaseParameters.valueAmount = VALUE_AMOUNT
purchaseParameters.valueCurrency = "VALUE_CURRENCY"

// Send the purchase conversion tag
HowtankWidget.sharedWidget().conversion(withType: "GOAL_NAME",
                                       andPurchaseParameters: purchaseParameters)
```

Where:

Name	Description	Mandatory?
GOAL_NAME	String, the name of the goal you want to track. You can have several goals (e.g. purchase, registration, subscription, etc.). Our platform will automatically generate a report for each goal tracked.	Yes
IS_NEW_BUYER	true : this is the first purchase of this user false : this user is a returning customer	Yes
PURCHASE_ID	String, identifier of the purchase in your system	Yes
VALUE_AMOUNT	The amount of the purchase	Yes
VALUE_CURRENCY	ISO Currency code string (eg. USD, EUR, GBP, etc.)	Yes