

# AtoneCon iOS SDK

The AtoneCon iOS SDK make it easy to perform an Atone payment inside your iOS app.

## A. Requirements

---

- iOS 8.0+
- Xcode 8.3

## B. Installation

---

### 1. Use CocoaPods

Embedded frameworks require a minimum deployment target of iOS 8

#### CocoaPods (recommended)

[CocoaPods](#) is a dependency manager for Cocoa projects. You can install it with the following command:

```
$ gem install cocoapods
```

CocoaPods 1.2+ is required to build AtoneCon 1.0+

To integrate AtoneCon into your Xcode project using CocoaPods, specify it in your `Podfile`:

```
source 'https://github.com/CocoaPods/Specs.git'
platform :ios, '8.0'
use_frameworks! # swift project

pod 'AtoneCon', '~> 1.0'
```

Then, run the following command:

```
$ pod install
```

### 2. Manual

- Open up Terminal, cd into your top-level project directory, and run the following command "if" your project is not initialized as a git repository:

```
$ git init
```

- Add AtoneCon as a git submodule by running the following command:

```
$ git submodule add git@github.com:AsianTechInc/AtoneCon-iOS.git
```

- Open the new AtoneCon folder, and drag the AtoneCon.xcodeproj into the Project Navigator of your application's Xcode project.

It should appear nested underneath your application's blue project icon. Whether it is above or below all the other Xcode groups does not matter.

- Select the AtoneCon.xcodeproj in the Project Navigator and verify the deployment target matches that of your application target.
- Next, select your application project in the Project Navigator (blue project icon) to navigate to the target configuration window and select the application target under the "Targets" heading in the sidebar.
- In the tab bar at the top of that window, open the "General" panel.
- Click on the + button under the "Embedded Binaries" section.
- You will see two different AtoneCon.xcodeproj folders each with two different versions of the AtoneCon.framework nested inside a Products folder.

It does not matter which Products folder you choose from, but it does matter whether you choose the top or bottom AtoneCon.framework.

- Select the top AtoneCon.framework for iOS
- The AtoneCon.framework is automatically added as a target dependency, linked framework and embedded framework in a copy files build phase which is all you need to build on the simulator and a device.

## C. Usage

---

### 1. Configuration

```
var options = AtoneCon.Options()
options.publicKey = "bB2uNvcOP2o8fJzHpWUumA"
let atoneCon = AtoneCon.shared
atoneCon.config(options)
atoneCon.delegate = self // AtoneConDelegate
```

## 2. Perform payment

### Create new payment

```
// These are required properties

var payment = AtoneCon.Payment(
    amount: 10,
    shopTransactionNo: "",
    checksum: "iq4gHR9I8LTszpozjDIaykNjuIsYg+m/pR6JFKggr5Q="
)

/**
The following attributes are not required.
If the attribute has value, it must be passed to the object.
If the property has no value, then set nil or not is mentioned.
*/

payment.salesSettled = false // Bool?
payment.descriptionTrans = "備考です。" // String?
```

### Configure customer

#### Create new customer

```
// These are required properties
var customer = AtoneCon.Customer(name: "接続テスト")

/**
The following attributes are not required.
If the attribute has value, it must be passed to the object.
If the property has no value, then set nil or not is mentioned.
*/

customer.nameKana = "セツゾクテスト" // String?
customer.companyName = "(株) ネットプロテクションズ" // String?
...
...
```

### Configure customer

```
payment.customer = customer
```

### Configure destination customers ( The attribute are not required )

#### Create destination customer

```
// These are required properties
var desCustomer = AtoneCon.DesCustomer(
    name: "銀座太郎",
    zipCode: "123-1234",
    address: "東京都中央区銀座 1 - 1 0 - 6   銀座ファーストビル4階"
)

/**
The following attributes are not required.
If the attribute has value, it must be passed to the object.
If the property has no value, then set nil or not is mentioned.
*/

desCustomer.nameKana = "ぎんざたろう" // String?
desCustomer.companyName = "株式会社ネットプロテクションズ" // String?
...
```

### Configure destination customers

```
payment.desCustomers = [desCustomer]
```

## Configure shop items

### Create items

```
// These are required properties
var item = AtoneCon.Item(
    id: "1",
    name: "1 0円チョコ",
    price: 10,
    count: 1
)

/**
The following attributes are not required.
If the attribute has a value, it must be passed to the object.
If the property has no value, then nil or not is mentioned.
*/

item.url = "https://atone.be/items/1"
```

### Configure shop items

```
payment.desCustomers = [item]
```

## Perform a payment

```
AtonePay.performPayment(payment)
```

## 3. Handle payment delegation

```
extension Controller: AtoneConDelegate {  
    func atoneCon(atoneCon: AtoneCon, didReceivePaymentEvent event: AtoneCon.PaymentEvent) {  
        switch event {  
        case .authenticated(let authToken):  
            // return authToken  
        case .cancelled:  
            // payment did cancelled  
        case .failed(let response):  
            // payment did failed  
            // response type [String:Any]  
        case .finished(let response):  
            // payment did finised  
            // response type [String:Any]  
        case .error(let error):  
            // payment did occur error  
        }  
    }  
}
```