# ucollect-ios

U-collect in-app payment SDK for iOS.

##Installation

####CocoaPods

Add the following line to your pod file

```pod

pod 'Ucollect','~> 1.0.2'

```

##Usage

###Initialization

```swift

var requestManager : RequestManager!

self.requestManager = try? RequestManager.initialize(context: self, merchantID: merchantID, merchantKey: merchantKey)

```

###Testing Mode

To activate testing mode

Add the following line after initializing the RequestManager

```swift

requestManager.workingMode = .DEBUG; // For Test

```

Add this exception to your info.plist file.

```xml

<key>NSAppTransportSecurity</key>

<dict>

<key>NSAllowsArbitraryLoads</key>

<true/>

</dict>

```

Remove the exception when testing is complete.

###Building Transaction Request

To start a transaction, let your ViewController implement the TransactionCallback protocol

```swift

let currentDate = Date()

let dateFormatter = DateFormatter()

dateFormatter.dateFormat = RequestManager.UCOLLECT\_DATE\_FORMAT

let convertedDate: String = dateFormatter.string(from: currentDate)

requestManager.transactionDateTime = convertedDate;

//Customer Info

requestManager.customerLastName = "Aleke";

requestManager.customerFirstName = "Godwin";

requestManager.customerEmail = "test@test.com";

requestManager.customerPhoneNumber = "08085555643";

//Payment Info

requestManager.countryCurrencyCode = "566";

requestManager.totalPurchaseAmount = 50000.0;

requestManager.numberOfItems = 5;

requestManager.purchaseDescription = "Buns Purchase";

requestManager.merchantGeneratedReferenceNumber = "\((UInt)(NSDate().timeIntervalSince1970 \* 1000))"

//Card Details

requestManager.cardPan = cardNumberText.text!

requestManager.cardCVV = cvvText.text!

requestManager.cardExpiryMonth = Int(expiryMonthText.text!)!

requestManager.cardExpiryYear = Int(expiryYearText.text!)!

requestManager.cardHolderName = cardHolderNameText.text!

if let pin = pinText.text{

requestManager.cardPin = pin

}

requestManager?.startPaymentTransaction(transactionCallback: self)

```

###Authorizing Transactions

When a transaction needs to be authorized using OTP, implement the onRequestAuthorization, and call requestManager.authorizeTransaction(otp);

```swift

…

func onRequestOtpAuthorization() {

showProgress(show: false)

let inputController = UIAlertController(title: "Authorization Request", message: "Enter OTP", preferredStyle: .alert)

inputController.addTextField { (textField) in

textField.keyboardType = .numberPad

textField.returnKeyType = .done

textField.placeholder = "Enter OTP"

}

inputController.addAction(UIAlertAction(title: "Authorize", style: .default, handler: { (action) in

self.showProgress(show: true)

self.requestManager.authorizeTransaction(otp: (inputController.textFields?[0].text!)!)

}))

inputController.addAction(UIAlertAction(title: "Cancel", style: .cancel, handler: { (action) in

self.showProgress(show: true)

self.requestManager.queryTransactionStatus(merchantGeneratedReferenceNumber: self.requestManager.merchantGeneratedReferenceNumber, resultCallback: self)

}))

self.show(inputController, sender: nil)

}

```

###Querying Transaction Status

To query the status of an on-going or already complete transaction

```swift

String merchantGeneratedReferenceNumber = "14811308291201"; // Previous Transaction's Merchant Generated Reference Number

self.requestManager.queryTransactionStatus(merchantGeneratedReferenceNumber: "14811308291201", resultCallback: self);

```