

# Integration Guide

## AuroPay iOS SDK

*VERSION 0.1*

*Aurionpro Payment – Trusted partner to solve your payment problems, while you focus on your business*

*Personalised experience for your customers for Faster receivables*

Aurionpro Solutions Limited | Synergia It Park, Plot No-R-270, Navi Mumbai, Maharashtra 400701

# Table of Contents


**1. Introduction ..... 4**

1.1 Prerequisites ..... 4

1.2 Retrieve API Credentials ..... 4

1.3 Integration Steps ..... 7

1.1.3 Install the SDK ..... 7

1.2.3 Configure Permissions ..... 8

1.3.3 Using the SDK ..... 8

1.4.3 Initialize the SDK ..... 9

1.4 Handling Payments ..... 9

1.1.4 Initiate a Payment ..... 9

1.2.4 Handle Payment Callbacks ..... 10

1.5 Troubleshooting ..... 10

**2. Need Help? ..... 11**

# RECORD OF REVISIONS

UAT All Versions:

| Version No. | Release Date | Features |
|---|---|---|
| **1.1.4** | 7-01-2025 | Bug fixes and Performance Improvement |
| **1.1.3** | 21-10-2024 | • Support for multi-Currency<br>• Changed card OTP collection flow for Singapore and Indonesia<br>• Removed RBI guidelines badge for countries other than India.<br>• Domestic and International card validation Saved card UI improvements |
| **1.1.2** | 19-09-2024 | Performance improvements and minor bug fixes |
| **1.1.1** | 13-09-2024 | PCI DSS Compliance changes. |
| **1.1.0** | 12-10-2023 | Save card feature Collect additional user info form |
| **1.0.1** | 14-07-2023 | Base release |

PROD All Versions:

| Version No. | Release Date | Changes |
|---|---|---|
| **1.1.3** | 10-01-2025 | Bug fixes and Performance Improvement |
| **1.1.2** | 23-10-2024 | • Support for multi-Currency<br>• Changed card OTP collection flow for Singapore and Indonesia<br>• Removed RBI guidelines badge for countries other than India.<br>• Domestic and International card validation Saved card UI improvements |
| **1.1.1** | 09-10-2024 | PCI DSS Compliance changes. |
| **1.1.0** | 13-10-2024 | Save card feature Collect additional user info form |
| **1.0.0** | 14-07-2023 | Base release |

# AuroPay iOS SDK Integration Guide

## 1. Introduction

The AuroPay Payments iOS SDK enables seamless payment processing in your mobile application. This guide walks you through integrating the AuroPay SDK into your iOS application.

### 1.1 Prerequisites

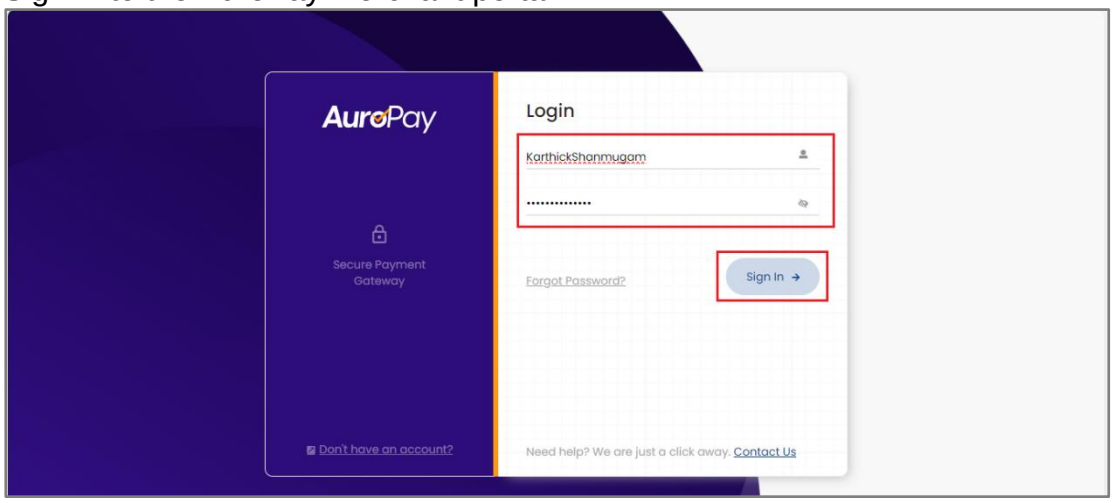
Before integrating the AuroPay SDK, ensure you have the following:

- A registered account on the AuroPay merchant portal.
- Merchant portal credentials for login.
- You need a **Subdomain ID**, **Access Key**, and **Secret Key** (available on the AuroPay Merchant Portal).

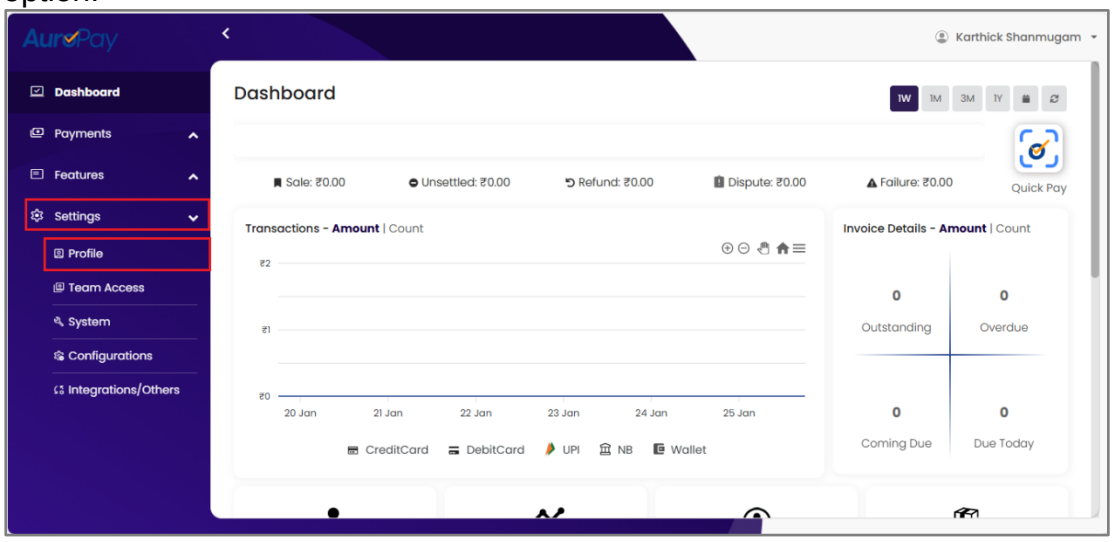
### 1.2 Retrieve API Credentials

To obtain the Subdomain ID, Access Key, and Secret Key, follow these steps:

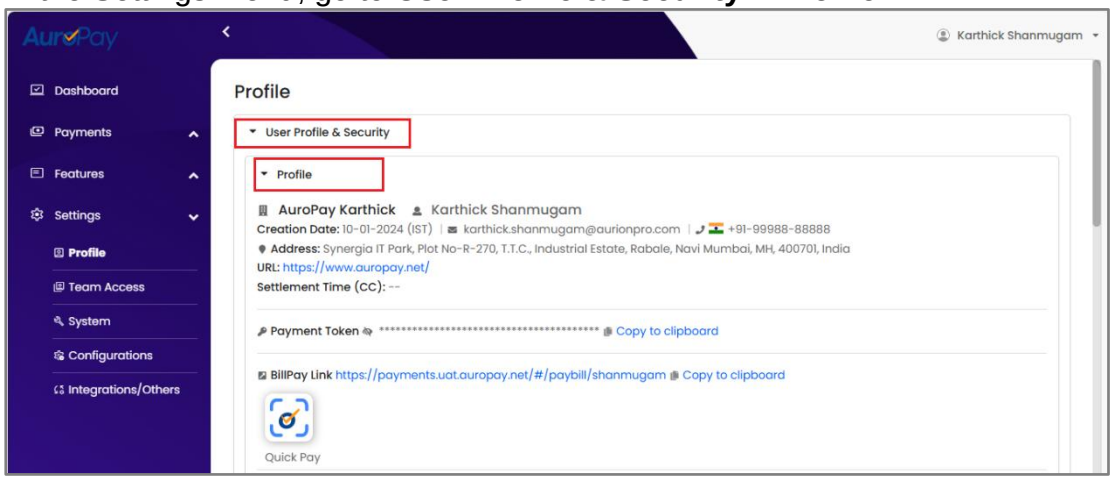
1) Sign in to the AuroPay merchant portal.



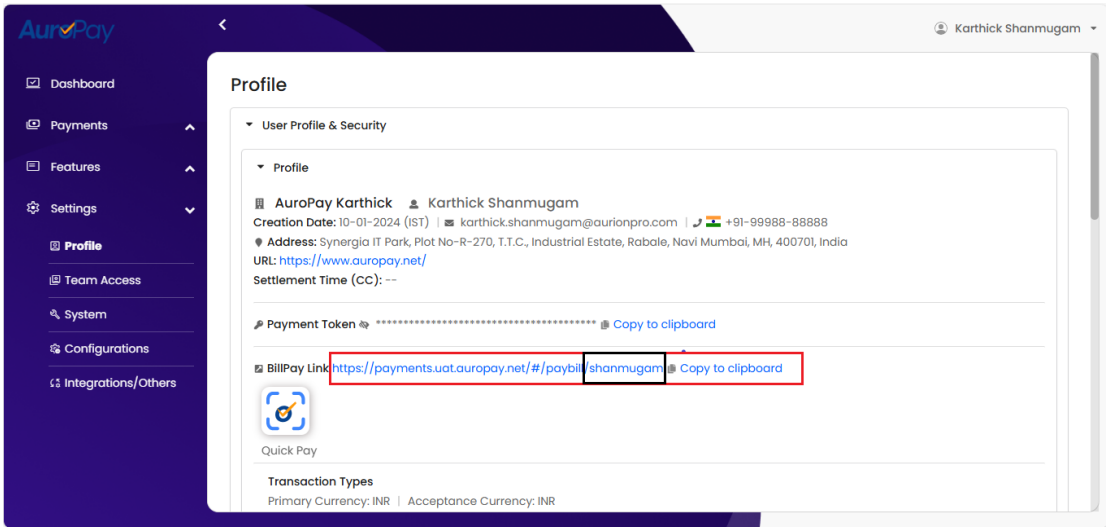
2) Within the AuroPay portal, go to the **Settings** dropdown and click the **profile** option.



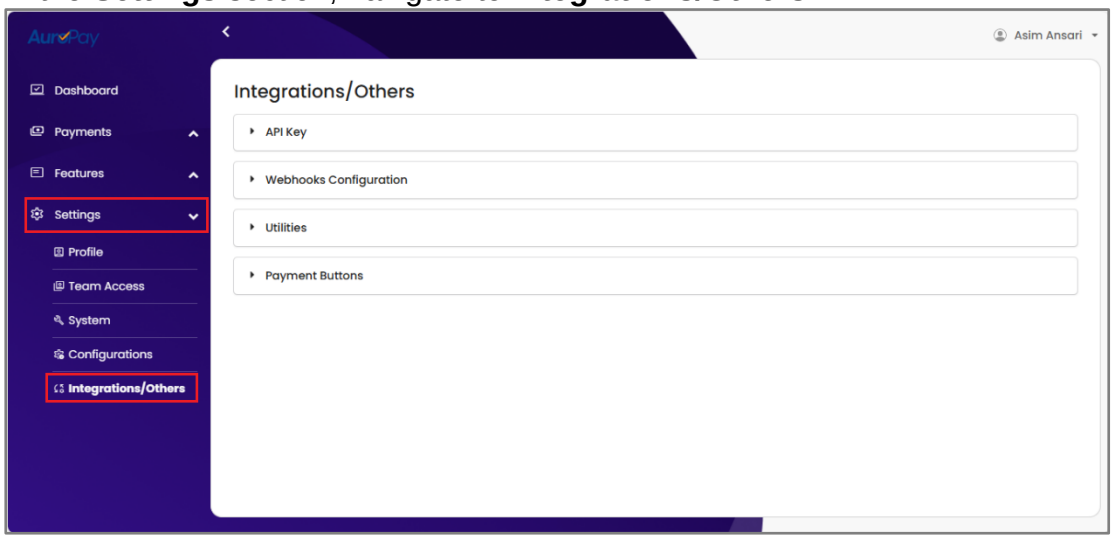

3) In the Settings menu, go to **User Profile & Security** > **Profile**.



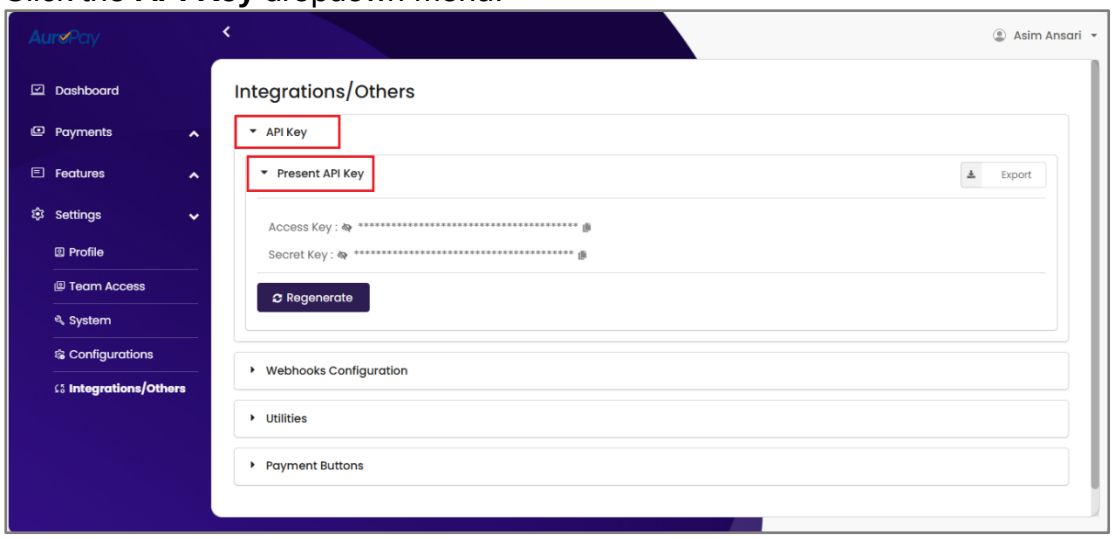

4) In the **profile** section you can view your **Subdomain ID** next to **BillPay Link** text.



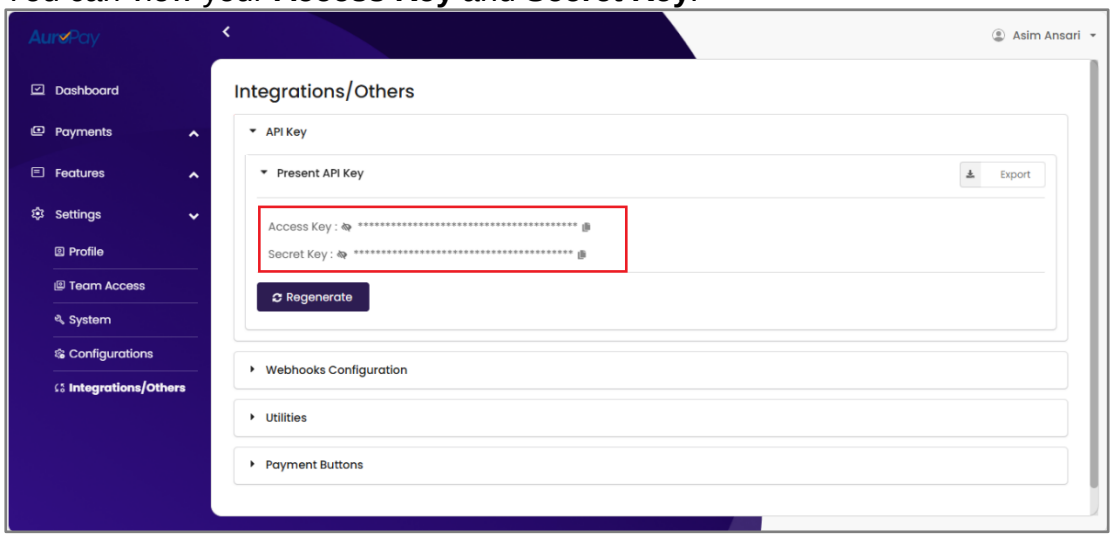
5) In the **Settings** section, navigate to **Integrations/Others**.




6) Click the **API Key** dropdown menu.




7) You can view your **Access Key** and **Secret Key**.



**Note: -** *Do not hardcode your API and access keys in your application. Always store them securely to protect against unauthorized access.*

## 1.3 Integration Steps

**SDK Requirements**

- **Minimum Deployment Target:** iOS 13 and above.
- **Recommended Target:** Latest available versions

### 1.1.3 Install the SDK

Choose one of the following methods:

**CocoaPods**

To add the AuroPay Payments iOS SDK to your project

1) Open your **Podfile** and add the following:

**For UAT:**

```
pod 'auropay-payments-sandbox'
```

**For Production:**

```
pod 'auropay-payments'
```

2) Save and close the **Podfile**.

3) Navigate to your project's root directory and run:

```
pod install
```

4) Open your **.xcworkspace** file. The AuroPay SDK should now be added to your project.

**Using Swift Package Manager**

1) In Xcode, select **File** > **Add Packages**.

2) Enter the repository URL:

- **For UAT:** https://github.com/Auropay/Sandbox-Auropay-Payments-SPM
- **For Production:** https://github.com/Auropay/Auropay-Payments-SPM

3) Select the latest version number.

### 1.2.3 Configure Permissions

The AuroPay SDK provides a **card scanning feature** for user convenience. To enable this feature, add the following key to the **Info.plist** file:

```
<key>NSCameraUsageDescription</key>
<string>Allow access to scan debit/credit cards.</string>
```

If this key is missing, the card scanning feature will not work.

### 1.3.3 Using the SDK

To process transactions with the AuroPay Payments iOS SDK, initialize the SDK before initiating a payment request.

**Import the SDK**

Add the required import statement:

**Swift**

```
// UAT
import AuroPayPaymentsSandbox

// Production
import AuroPayPayments
```

**Objective-C**

```
// UAT
@import AuroPayPaymentsSandbox;
or
#import <AuroPayPaymentsSandbox/AuroPayPaymentsSandbox-Swift.h>

// Production
@import AuroPayPayments;
or
#import <AuroPayPayments/AuroPayPayments-Swift.h>
```

### 1.4.3 Initialize the SDK

**Swift**

```
let apb = AuroPayBuilder()
    .merchantID("Your Subdomain ID")
    .accessKey("Your Access Key")
    .secretKey("Your Secret Key")
    .showReceipt(true)
    .allowCardScan(true) // Optional
    .build()
```

**Objective-C**

```
AuroPay *apb = [[[[AuroPayBuilder alloc]
    merchantID:@"Your Subdomain ID"]
    accessKey:@"Your Access Key"]
    secretKey:@"Your Secret Key"]
    showReceipt:YES]
    allowCardScan:YES]
    build];
```

## 1.4 Handling Payments

### 1.1.4 Initiate a Payment

Call **doPayment()** to start a transaction.

**Swift**

```
apb.doPayment(displayViewController: self, amount: 100.00, andDelegate: self)
```

**Objective-C**

```
[apb doPaymentWithDisplayViewController:self amount:100.00 andDelegate:self referenceNumber:@"" askCustomerDetails:NO];
```

### 1.2.4 Handle Payment Callbacks

Implement the following delegate methods:

**Swift**

```
extension ViewController: APPaymentCompletionProtocol {
    func onPaymentSuccess(_ orderId: String, transactionStatus: Int, transactionId: String) {
        print("Payment successful: Order ID - \(orderId), Transaction ID - \(transactionId)")
    }

    func onPaymentError(_ message: String) {
        print("Payment failed: \(message)")
    }
}
```

**Objective-C**

```
- (void)onPaymentSuccess:(NSString *)orderId transactionStatus:(NSInteger)transactionStatus transactionId:(NSString *)transactionId {
    NSLog(@"Payment successful: Order ID - %@, Transaction ID - %@", orderId, transactionId);
}

- (void)onPaymentError:(NSString *)message {
    NSLog(@"Payment failed: %@", message);
}
```

## 1.5 Troubleshooting

| Issue | Solution |
|---|---|
| **SDK not found after CocoaPods installation** | Run **pod install --repo-update** |
| **Payment fails with API key error** | Ensure keys are retrieved from the backend, not hardcoded |
| **Card scan not working** | Check if **NSCameraUsageDescription** is added to **Info.plist** |

## 2. Need Help?

If you encounter any issues, reach out to our support team at support@auropay.net.