

YOUR  
TRUSTED  
PARTNER



# AYAPay

**Online Merchant  
Integration Guide**

**Nov 2020**

**Team**

Aya Innovation Labs

## 1. Accounts Setup

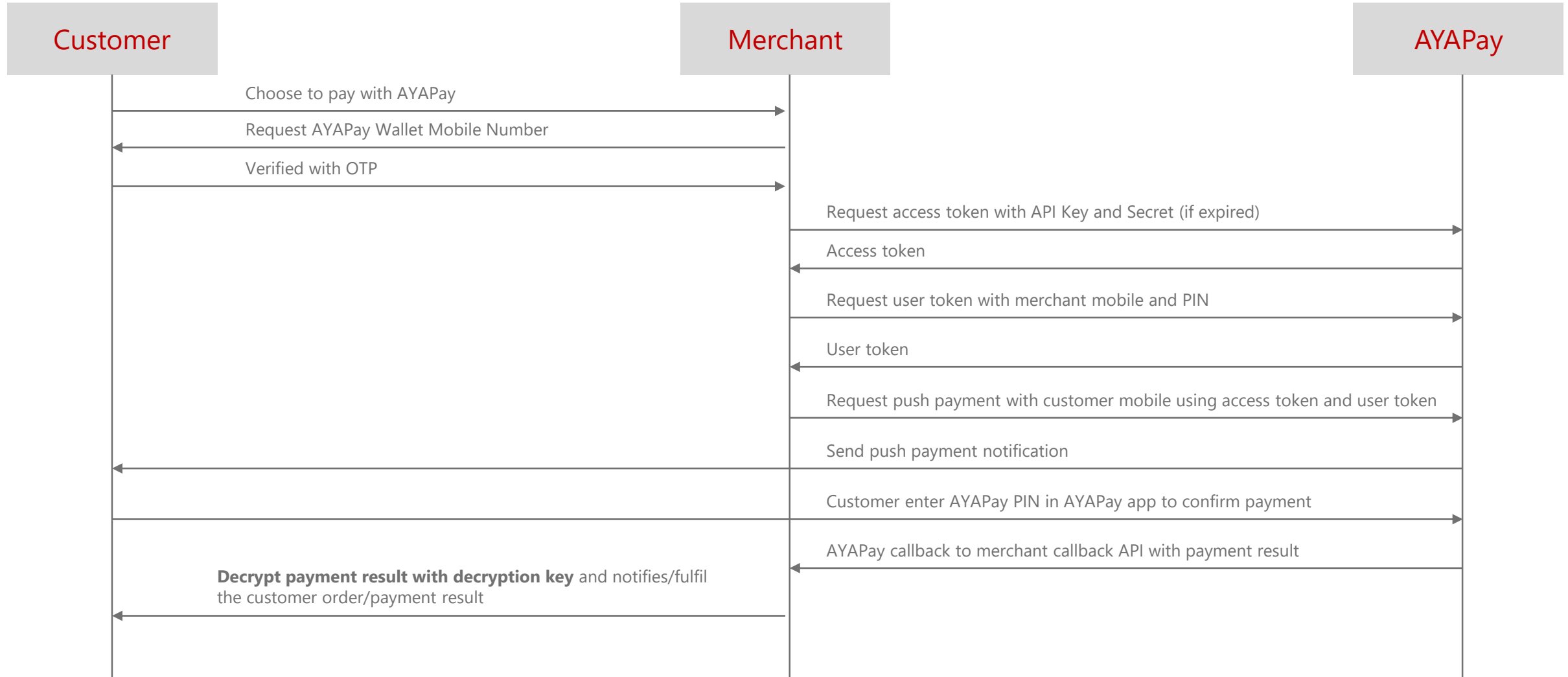
1. Developer account
2. Merchant account

## 2. Environment Setup

1. Dev. APKs
  - Subscriber app + account
  - Merchant app
2. Postman collection, environment

## 3. Integration

1. Access token
2. User token
3. Payment request
  1. Push payment
  2. QR payment
4. Refund request



## Prerequisites

- Visual Studio 2019
- MS SQL – the demo used MS SQL but you can use any database
- Ngrok – to test callback URL with tunneling to your local PC (alternatively you can deploy on your test server and provide AYAPay with the callback URL)
- RestSharp for API requests
- Newtonsoft for json deserialization
- **AYAPay** – API key and secret, merchant account (mobile, PIN), decryption key for callback URL endpoint
- Development APKs – AYAPay subscriber, AYAPay merchant

# How to request push payment?

## Request Push Payment

1. Copy **AyaPayDotNet** folder into your project and import the necessary nuget packages (RestSharp, Newtonsoft, etc.)
2. In **AyaPay.cs** – update your API key, secret, merchant account mobile number, PIN and decryption key provided
3. Call **RequestPushPayment** static method when you want to send push payment request to AYAPay user

## Callback API

In your callback URL endpoint, when you receive the callback from AYAPay:

1. Deserialize the response into **Callback** model
2. Decrypt the **PaymentResult** property of callback using **DecryptPaymentResponse** helper method
3. Refer **externalTransactionId** property of **PaymentResult** model to check against the transaction ID you passed to AYAPay when you request push payment
4. Update transaction, notify user accordingly

<https://developer.ayainnovation.com/devportal/apis>

<https://ngrok.com/>

<https://www.base64encode.org/>

<https://www.postman.com/>

<https://www.microsoft.com/en-us/sql-server/sql-server-downloads>

<https://visualstudio.microsoft.com/downloads/>

## Thank you!