



ARAKA PAYMENT MANUAL INTEGRATION

HOWTO USE PAYMENT PAGES FOR E-COMMERCE

Abstract

This document describes two ways of using ARAKA payments platform for website e-commerce

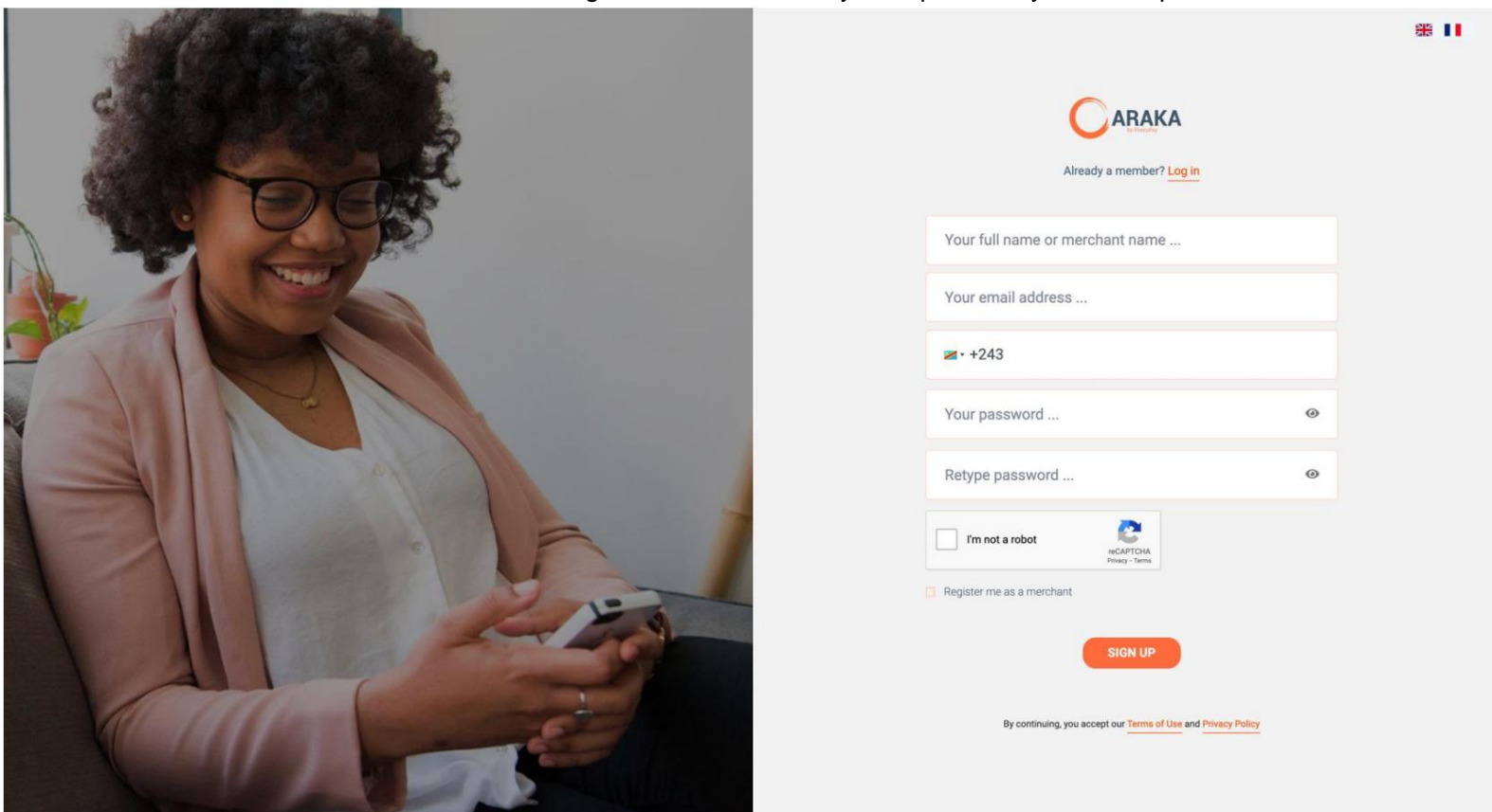
Contents

- 1. Create an Araka Business Account.....2
- 2. Collect Payments.....3
 - 2.1 One-time Payment Page 4
 - 2.2 E-COMMERCE SHOPPING CART..... 5
 - 2.3 ANDROID APP INTEGRATION 6
 - 2.4 iOS APP INTEGRATION..... 6



1. Create an Araka Business Account

As a merchant looking to use the Araka Payment platform, you are required to



The image shows a woman with curly hair and glasses, smiling and holding a smartphone. To her right is the Araka merchant registration form. The form includes the Araka logo, a 'Log in' link for existing members, and input fields for 'Your full name or merchant name ...', 'Your email address ...', a phone number (prefixed with '+243'), 'Your password ...', and 'Retype password ...'. There is a reCAPTCHA checkbox labeled 'I'm not a robot' and a 'Register me as a merchant' checkbox. A 'SIGN UP' button is at the bottom. A footer note states: 'By continuing, you accept our [Terms of Use](#) and [Privacy Policy](#)'.

create a merchant/business account on Araka. Go to <https://www.arakapay.com/auth/register> to create a merchant account

Figure 1. Araka Merchant Signup

Once you visit the URL above, perform the following steps:

1. Enter your Business Name
2. Enter your Email Address
3. Enter your Phone Number

4. Enter your Password and re-type to Confirm
5. Validate the re-Captcha
6. Select, "Register me as a merchant" checkbox
7. Click Sign Up

On completing the steps above you will receive a verification email in the email account you specified. This verification email contains a link that allows you to verify your account. Once you have verified your account, you are ready to start accepting payments with ARAKA.

2. Collect Payments

Once you've verified your merchant account you will have access to a Merchant Portal where you will see records of transactions made for your goods and services sold via ARAKA and the ability to create and manage payment pages. Go to <https://merchant.arakapay.com/> and login with your account's credentials.

With an ARAKA Merchant account, you can accept payments on for your goods and services in the following ways:



2.1 One-time Payment Page

A one-time payment page is the easiest way to collect payments prices without performing any code related integrations on your website. The one-time payment page can be accessed by a link which you can share with your customers via email,

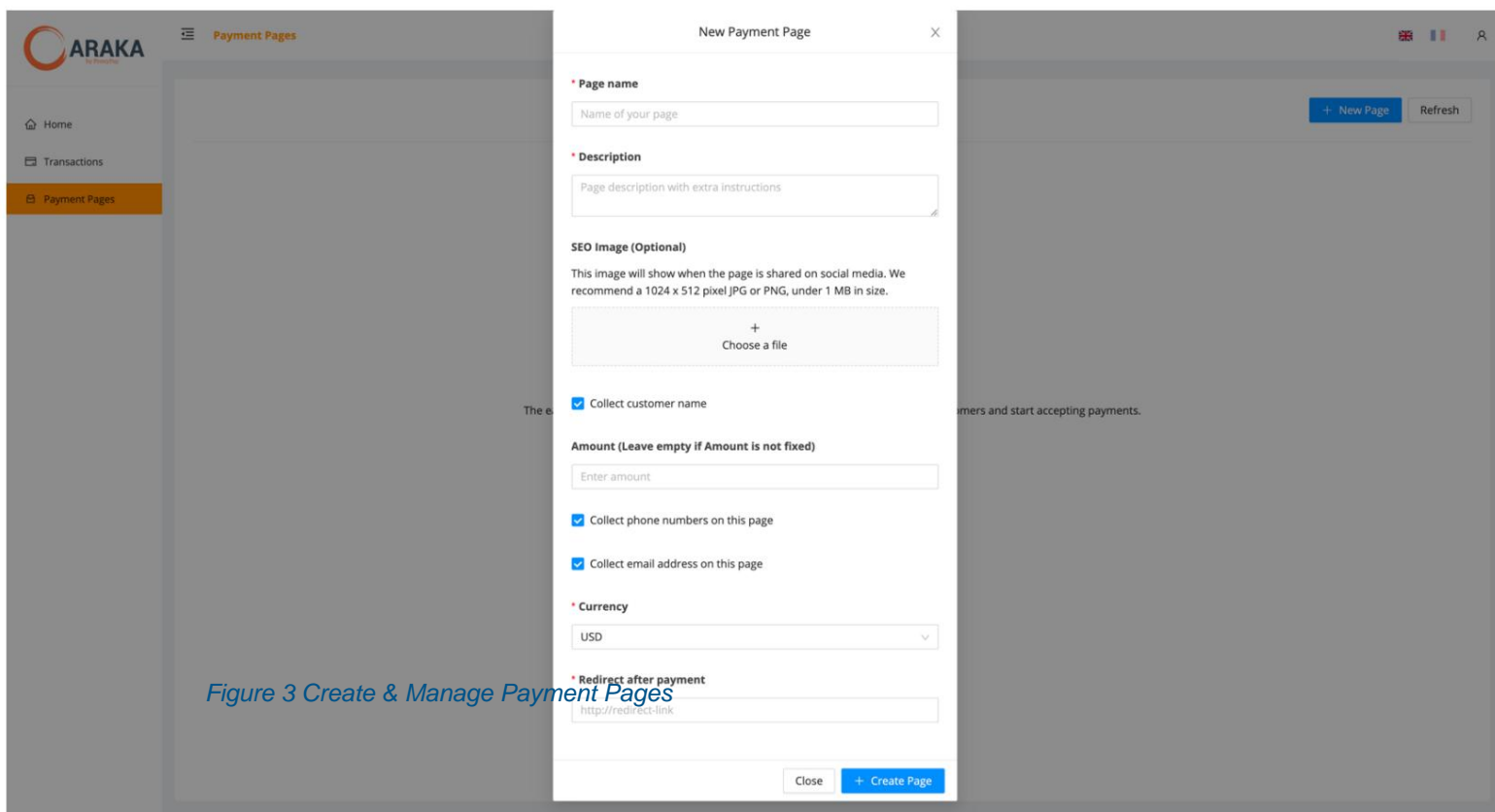


Figure 3 Create & Manage Payment Pages

WhatsApp, Twitter or any medium of your choice. This link is also unique to your business, and all your customer has to do is open it and pay you. Take the following steps to create a one-time payment page:

Figure 2 New One-time Payment Page

1. Login to the ARAKA Merchant Portal with the account you created
2. Navigate to Payment Pages located on the sidebar to the left of the screen
3. Click on "New Page"
4. From the option modal that appears, select "One-time Payment"
5. Fill in the details that are required and click "Create Page"
6. On the "Payment Page" screen (see Figure 4) you will see the page you just created.

7. Click on "View" to see the details of the payment page you just created.
8. Copy the Payment Page URL and share to your customers.

2.2 E-COMMERCE SHOPPING CART

For a much more custom feature, where you collect an amount depending on the customers choice of products, for example an e-commerce shopping cart, you can do the following with the one-time payment page:

1. Login to the ARAKA Merchant Portal with the account you created
2. Navigate to Payment Pages located on the sidebar to the left of the screen
3. Click on "New Page"
4. From the option modal that appears, select "One-time Payment"
5. Fill in the details that are required and click "Create Page"
6. Leave the "Amount" field empty
6. On the "Payment Page" you will see the page you just created.
7. Click on "View" to see the details of the payment page you just created.
8. Copy the Payment Page URL.
9. On your checkout page, when you are ready to collect payment, redirect the user with GET request to the "Payment Page" URL you copied.

10. Use the URL <https://merchant.arakapay.com/payment/AAAAA-BBBBBB-CCCCCDDDD> .

11. Include the following parameters as query strings in the URL:

```
"pageTitle": "Sample title",  
"pageDescription": "Sample description",  
"customerFullName": "Mike Wazowski",  
"customerPhoneNumber": "+2430990998098",  
"customerEmailAddress": "example@user.com",  
"transactionReference": "TRW344232134",  
"currency": "USD", "amount": "40", "merchant":  
"b255c76a-41cc-4398-bfd2-b2f9e3238c2d"
```

10. Once the transaction is completed or cancelled, the user is redirected to the URL provided in the "Redirect URL" field you specified when creating the Payment Page with the following sample payload:

```
{ "systemReference": "1234", "transactionStatus": "SUCCESS" }
```

The "systemReference" in this payload, can be identified as a "Transaction ID" on the Merchant portal, in the transactions table.

2.3 ANDROID APP INTEGRATION

When developing for Android, use the WebView activity to send the transaction payload to the payment page created (see section 2.2)

The transaction payload must include the properties as seen in Section 2.2

After the transaction has been successfully processed, the user will be redirected to the Application's Webview using the *“redirectURL”* that was provided when creating the payment page.

2.4 iOS APP INTEGRATION

When developing for iOS, use the WKWebView object and programmatically embed into your view hierarchy, or add it using Interface Builder. Send the transaction payload to the payment page created (see section 2.2)

The transaction payload must include the properties as seen in Section 2.2

After the transaction has been successfully processed, the user will be redirected to the Application's WKWebView object using the *“redirectURL”* that was provided when creating the payment page.

2.5 INTEGRATION API

For Merchants who prefer to use plain old API requests, there is the possibility of using our API.

The API requires a bearer token that can be obtained by using our login endpoint. After which a payment request is generated and the status of the payment is received via either a callback message or transaction status check API.

Details of the various requests are as follows:

2.5.1 Login

POST /api/login

Access to this endpoint requires setup changes on your account. Contact the technical team for this configuration to be completed

Request Formats

application/json, text/json

Sample:
{

```
"emailAddress": "string",
  "password": "string"
}
```

Response Information

application/json, text/json

Sample:

```
{
  "token": "string",
  "username": "your email"
}
```

2.5.2 Payment Request

POST /api/v1/pay/paymentrequest

Access to this endpoint requires setup changes on your account. Contact the technical team for this configuration to be completed

Request Formats

application/json, text/json

Sample:

```
{
  "order": {
    "paymentPagelId": "string",
    "customerFullName": "string",
    "customerPhoneNumber": "string",
    "customerEmailAddress": "string",
    "transactionReference": "string",
    "amount": 0,
    "currency": "string",
    "redirectURL": "string"
  },
  "paymentChannel": {
    "channel": "string", //MOBILEMONEY
    "provider": "string", //MPESA,AIRTEL
    "walletID": "string" //MSISDN
  }
}
```


Response Information

application/json, text/json

Sample:

```
{
  "transactionId": "string",
  "transactionReference": "string"
}
```

2.5.3 Status Inquiry

GET api/v1/reporting/transactionstatus/{transactionid}

Access to this endpoint requires setup changes on your account. Contact the technical team for this configuration to be completed

Request Formats

api/reporting/transactionstatus/xxxx

where xxxx is the transactionId received in the transaction request

Response Information

application/json, text/json

Sample:

```
{
  "transactionId": "string",
  "status": "string", //SUCCESS,FAILED
  "transactionDate": "string",
  "transactionDescription": "string" }
```

2.5.2 Send Mobile Money

POST /api/v1/pay/sendmobilemoney

Access to this endpoint requires setup changes on your account. Contact the technical team for this configuration to be completed

Request Formats

application/json, text/json

Sample:

```
{
  "order": {
    "paymentPagelId": "string",
    "customerFullName": "string",
    "customerPhoneNumber": "string",
    "customerEmailAddress": "string",
    "transactionReference": "string",
    "amount": 0,
    "currency": "string"
  },
  "destination": {
    "provider": "string", //MPESA,AIRTEL
    "walletID": "string" //MSISDN
  }
}
```

Response Information

application/json, text/json

Sample:

```
{
  "transactionId": "string",
  "transactionReference": "string"
}
```