

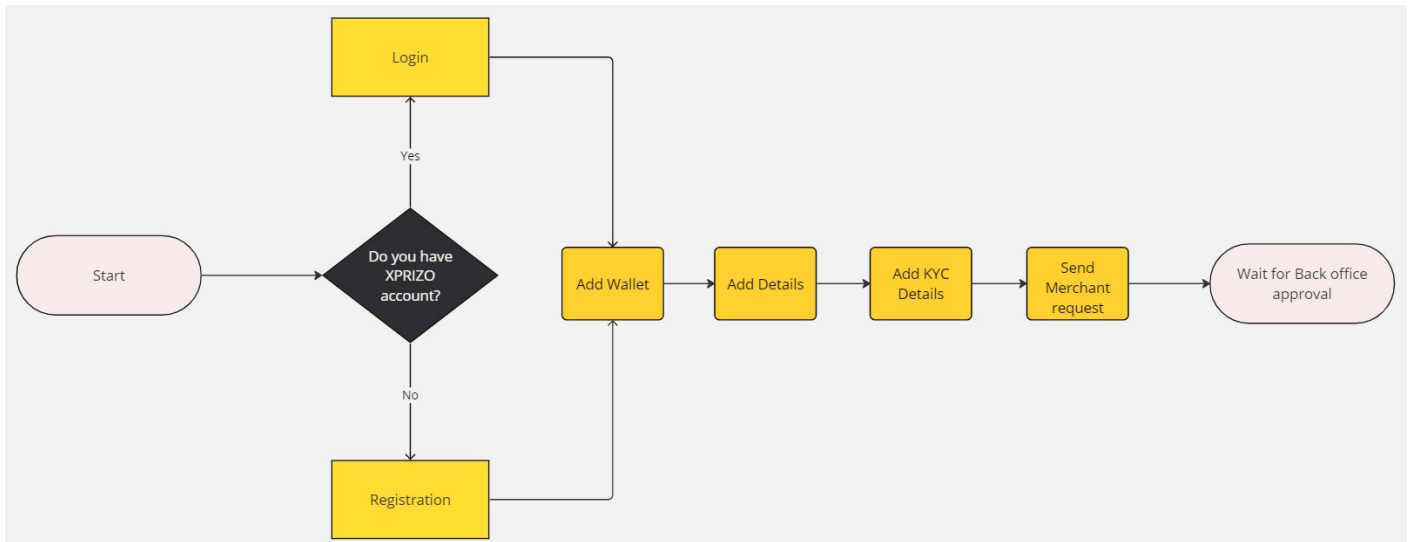


Merchant Manual

The Merchant Manual Documentation serves as a comprehensive guide tailored to empower merchants in effectively utilizing our platform. Covering a spectrum of topics, this manual offers detailed insights into account setup, payment processing, inventory management, and customer engagement strategies. Merchants will find user-friendly instructions complemented by illustrative examples to facilitate a smooth onboarding experience. The documentation dives into the intricacies of transaction handling, refund processes, and the utilization of promotional tools to maximize business growth. Additionally, it addresses common challenges, providing merchants with the knowledge needed to navigate potential hurdles seamlessly.

- [How to become a Merchant?](#)
- [How to do Integration?](#)
- [FAQs](#)
- [How to do Integration for Payment URL?](#)

How to become a Merchant?



1. Login/Registration

- To initiate the creation of a new account, please [click here](#) and follow the provided steps.

The image shows the XPRIZO user interface for login and registration. At the top is the XPRIZO logo. Below it is the 'Login' heading, followed by a subtext: 'Enter your username and password to login to your account, or select register if you are a new user.' On the left is a large, stylized XPRIZO logo graphic. To the right are two input fields: 'Login Account *' and 'Login Password *' (with a toggle icon). Below the password field is a 'Forgot Password' link. A large grey button with a right arrow and the text 'Login' is positioned below the inputs. Underneath this button is the text 'Not a Member ?' followed by a red-bordered 'Register' button. At the bottom left, it says 'or Login with :'. On the bottom right, there is a red circular button with a white 'G+' icon.

2. Add Wallet

To be able to perform any kind of transaction or manage funds the user needs to add a wallet in their profile to execute various transactional activities. By adding a wallet user can store their funds securely and carry out transactions directly from their account. To add a wallet follow the steps outlined below.

[Add a New Wallet](#)

3. Add User Details

To be able to send a merchant request to the back office, you will first need to complete user details for your profile. There you will need to complete your full name and address details. For a step-by-step guide, please click on the link below to learn how to add user details.

[User Details](#)

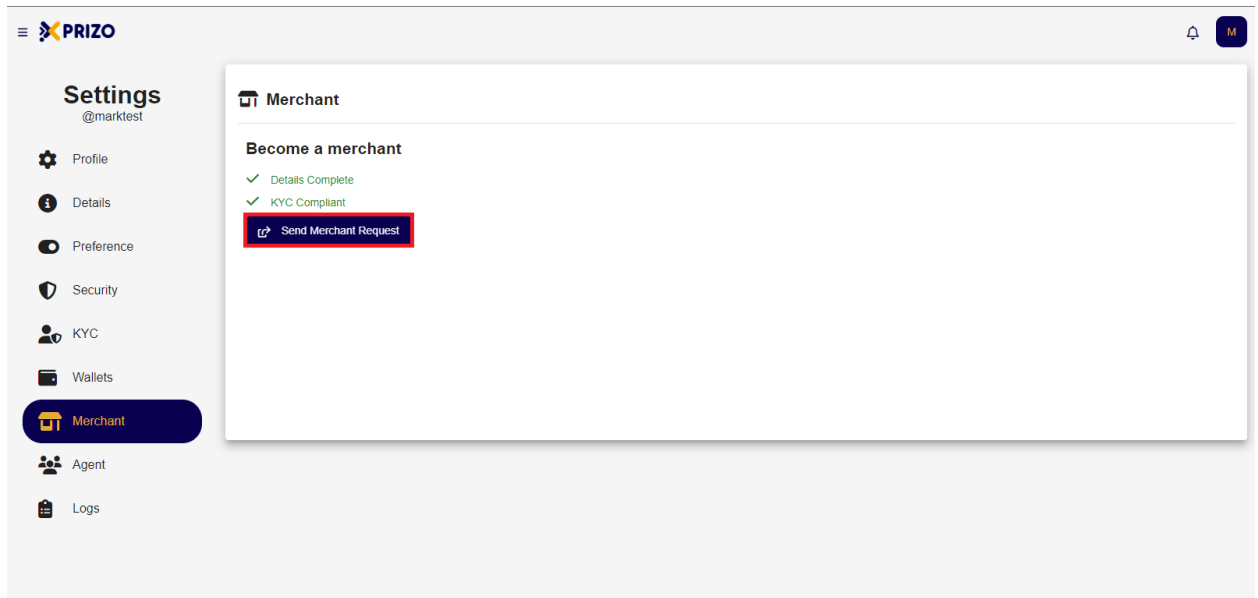
4. Add KYC Details

The next step is to complete KYC details and become KYC compliant. For this, you would have to upload your documents to verify your KYC. To initiate the KYC process, follow the provided steps below to complete the necessary procedures for KYC verification.

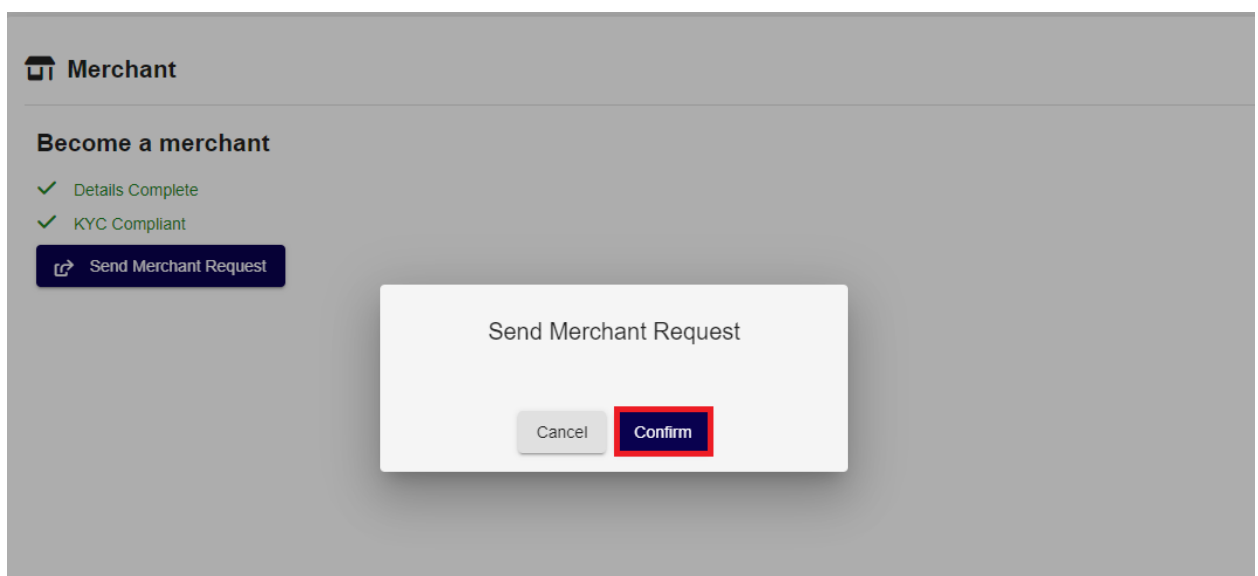
[KYC Details](#)

5. Send Merchant Request

- Navigate to the Merchant tab to verify if all the necessary steps are completed. Then, click on the "Send Merchant Request" to let the back office know you want to become a merchant.



- Click on the "Confirm" button to send the request.



- After sending the request, the user must wait for approval from the back office. Upon approval, the user will become a Merchant.

How to do Integration?

Introduction

This document provides detailed steps for setting up your merchant profile, creating a processing profile with access to the merchant wallet, and navigating through testing (sandbox) and production (live) environments. Additionally, it walks you through the deposit process using a card/ Xprizo wallet/ Mpesa/ UPI, covering the flow, validation, callbacks, deposit transaction creation, approval webhook set up, and status checking. Follow the sections below to manage your account and transactions efficiently.

Account Setup

You need to create two profiles where the first profile will be of the Merchant having a wallet to be able to receive funds and the other profile will be of Processing which will be used for creating the transactions on the merchant's account wallet.

Merchant profile

To create a merchant profile use the below-mentioned link where you can find how to do registration, profile updation, KYC, and procedure for merchant requests.

[How to Become a Merchant](#)

Processor profile

A processor profile is required to manage the transactions of the merchant wallet. **This processor profile will help the merchant, to secure his wallet by giving only the required access to the processor on the merchant profile.** To configure the processor profile use the below-mentioned link where you can find all the steps to configure the processor profile (linking of merchant profile and processor profile, and after that how to give access to the processor profile of the merchant profile).

**** You can also use a merchant profile as a processor profile, but this is advisable to have a separate processor profile to manage the transactions to make your**

Processor Profile Configuration

Test profile

To test the merchant account you will have to create a sample/test account which you will use to check if the integration is done properly or not.

Environment & Authentication

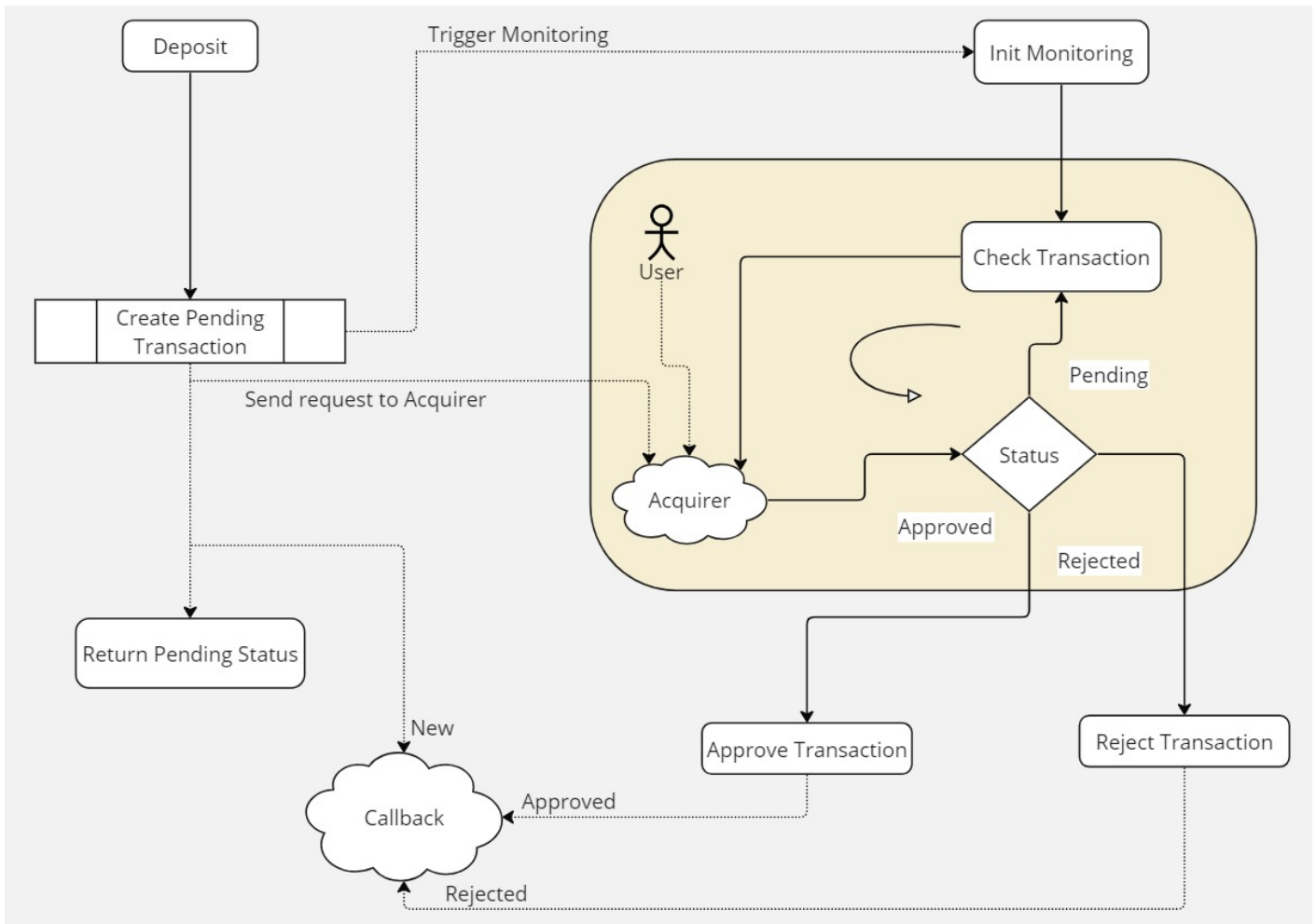
Before starting the integration, the user needs to know about the available environment and authentication methods required for integration. Use the below-mentioned link where you can find details about the available environment, and authentication methods (how to generate token & API key and usage of these).

Environment & Authentication

Deposits

Deposit Using Card

Flow



Steps

- Create Deposit Transaction Using Card
- Wait for a callback.
- Check status.

After you have called the **Transaction/CardDeposit** endpoint, a payment request is sent to the Acquirer network to process the card payment. A pending transaction is also created and Xprizo will continue monitoring the Acquirer network until an approval or rejection is received.

If the card is 'Approved' (2DS) then the transaction will be added to the merchant's wallet and the API will return an 'Active' status. No further action is required.

If there is a problem creating the transaction then the API will return a status of 'Rejected'.

It is possible that a card is not processed immediately such as some kind of delay or the user might need to approve the transaction using another device, then a 'Pending' status will be returned.

In the case of 3DS, there could be a redirect to an alternate URL. In this case, a 'Redirect' status and URL is returned.

The Merchant will get updates on the status of the transaction via callbacks. **To receive a callback, you will need to request to set this up on your Merchant account (the profile that is used to create this transaction).** This can be done by going to settings/preferences and setting the "Approval Webhook" to point to a URL of your choice. You can also set this using [Preference/SetApprovalWebhook](#)

A callback will be sent when

- The pending transaction is created.
- The pending transaction is approved
- The pending transaction is rejected

You can use [this link](#) to see the response coming on this Webhook.

**** You should, however, check the status, after you get a callback to confirm that the transaction was completed successfully. You can use [Transaction/Status](#) endpoint to check the status.**

Deposit Using Xprizo Wallet

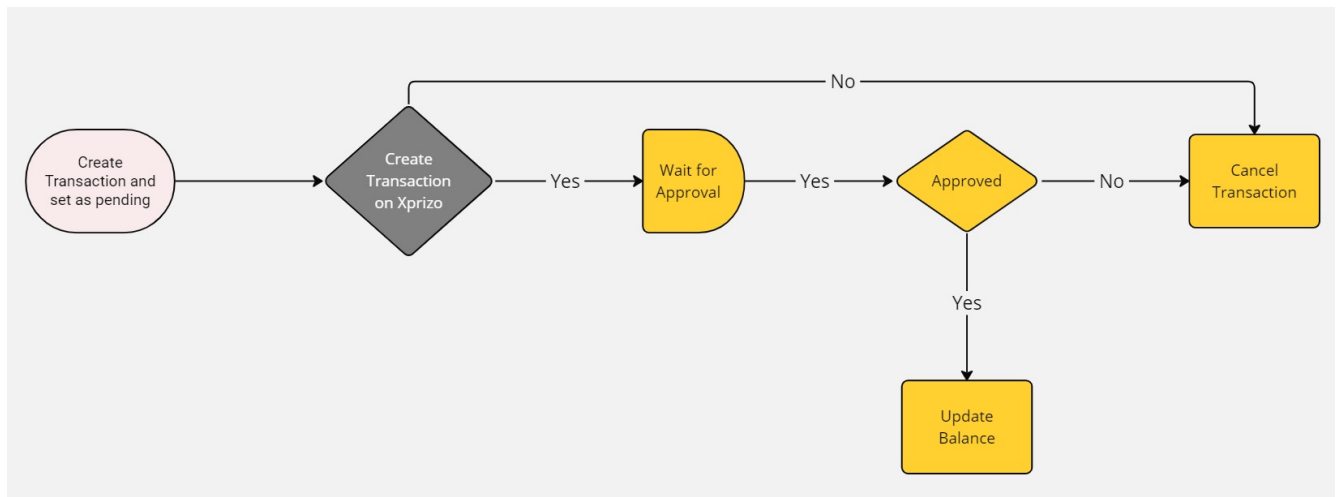
To make a deposit, funds are transferred from a user's wallet to the merchant's wallet. This process is done by requesting a payment from the user through the [Transaction/RequestPayment](#) endpoint.

Flow

This is an example of the flow that should be used when integrating with Xprizo to perform deposits.

- *(on merchant)* Create a new transaction.
 - Create a reference to be used for this transaction.
 - Set the transaction as pending.
- *(on Xprizo)* Create the Xprizo transaction.
 - Find the user's wallet using their phone number or email or username, and the specified currency through the [Wallet/Info](#) endpoint.
 - Find the merchant's wallet with the same currency using the [Wallet/Info](#) endpoint.
 - Create transaction using [Transaction/RequestPayment](#)

- Wait for the callback.
- Check the status of the transaction [Transaction/Status](#) endpoint.
- (on merchant) Update or cancel the transaction
 - If the transaction is approved, update the transaction status in your system.
 - If the transaction is rejected, cancel the transaction on your system.



Steps

- Create a payment Request
- Wait for callback
- Check Status

Once you've created a payment request, wait for a callback that will be sent once the user approves the request. While it's not mandatory to utilize callbacks, it's more efficient to do so. Alternatively, you can directly call the check status endpoint, but this isn't recommended as you would need to continuously poll the server to check the status. Use this endpoint to check the status [Transaction/Status](#).

To receive callbacks, you must set this up on your account (the profile that is used to create this transaction). You can set up the callback by accessing the settings/preferences in Xprizo. Configure the "Approval Webhook" to your desired API endpoint where you want to receive the callback.

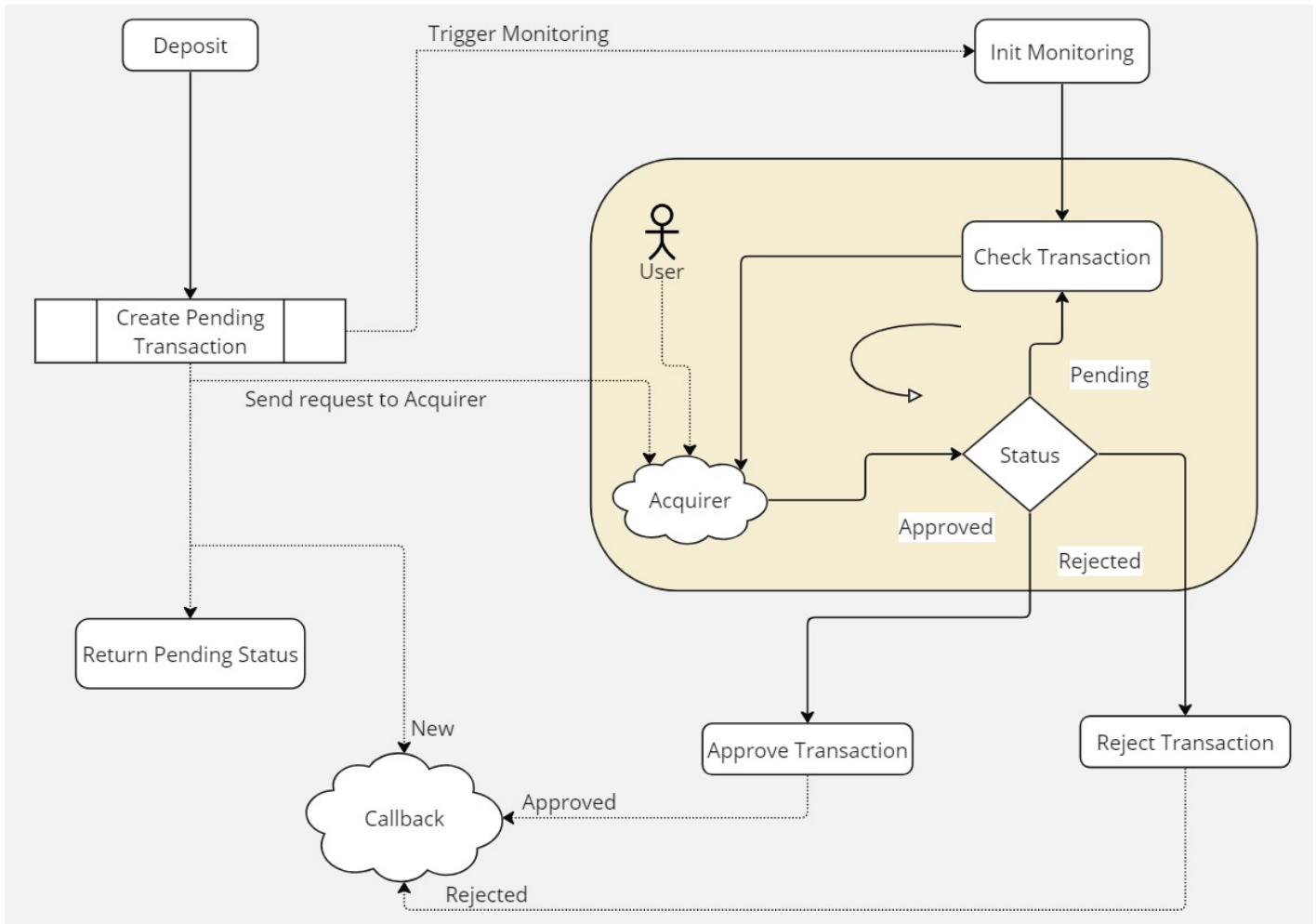
Alternatively, you can set this up using the [Preference/SetApprovalWebhook](#) endpoint.

**** You should, however, check the status, after you get a callback to confirm that the transaction was completed successfully. You can use [Transaction/Status](#)**

Deposits Using M-Pesa

To make deposits from a user via M-Pesa, initiate a deposit by transferring funds from the user's M-Pesa account to your merchant's wallet using the [Transaction/MPesaDeposit](#) endpoint.

Flow



Steps

- Add an M-Pesa deposit Transaction
- Wait for callback
- Check Status

Once you've called the "MpesaDeposit" endpoint, a payment request is sent to the M-Pesa network and the user will then confirm the transaction on their phone. A pending transaction is also

generated, and Xprizo will continuously monitor the M-Pesa network until a confirmation or failure is received.

- If a confirmation is received, the pending transaction will transition into a completed transaction.
- If the transaction is cancelled, a notification containing details about the cancellation will be generated and sent to the merchant's account.

Merchants will receive updates on the transaction status via callbacks. To enable callbacks, you'll need to configure this on your account (the profile used to create the transaction). You can accomplish this by navigating to the settings/preferences in Xprizo and configuring the "Approval Webhook" to point to a URL of your choice. Alternatively, you can achieve the same using the

[Preference/SetApprovalWebhook](#) endpoint.

A callback will be sent when

- the pending transaction is created.
- the pending transaction is approved
- the pending transaction is cancelled

It is not strictly necessary to use callbacks, you have the option to periodically check the status of the transaction. However, it's not recommended as it would require to poll the server continuously to check the status. However, it's advisable to verify the status after receiving a callback to ensure that the transaction was completed successfully.

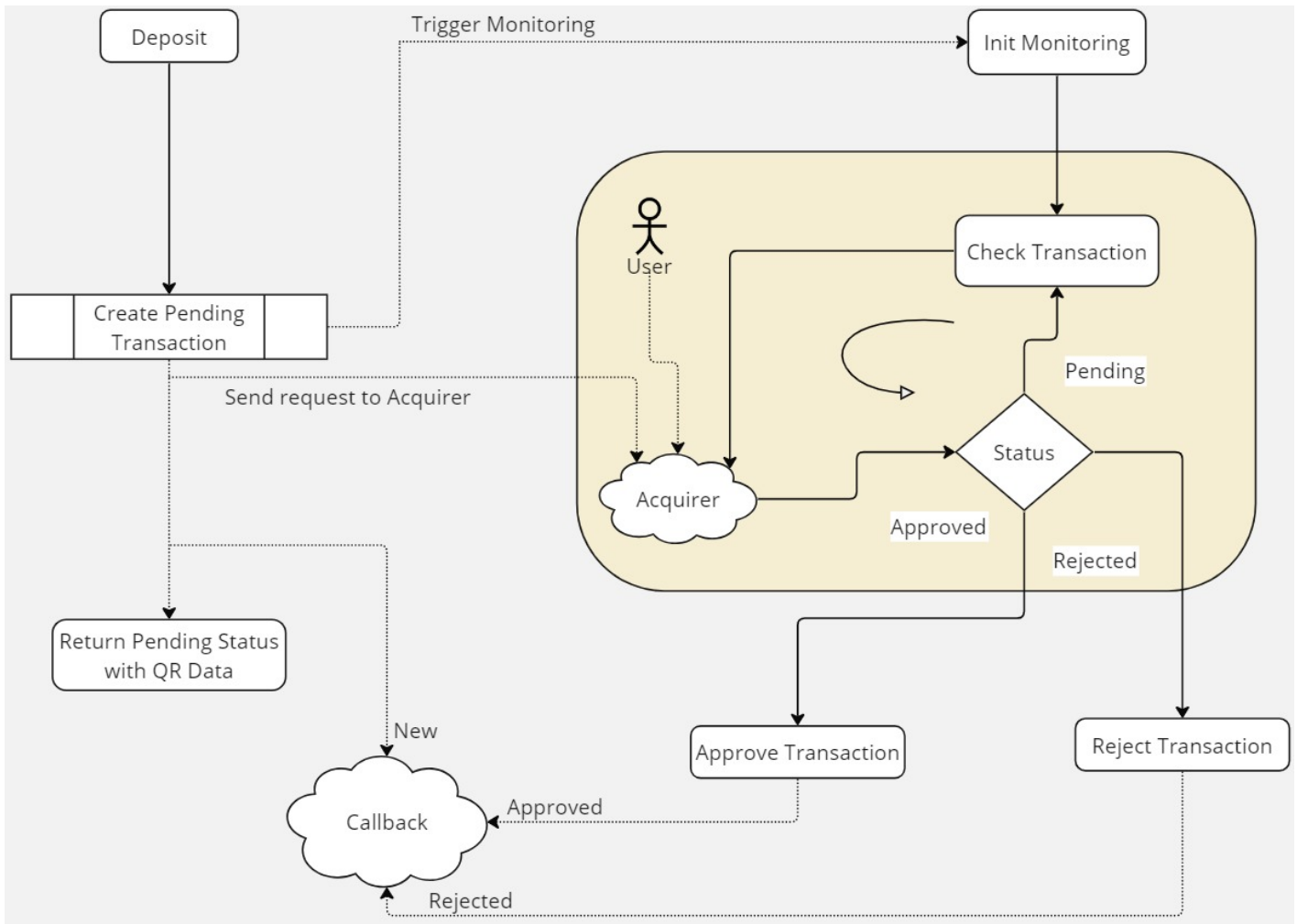
Use this endpoint to check the status [Transaction/Status](#).

**** You should, however, check the status, after you get a callback to confirm that the transaction was completed successfully. You can use [Transaction/Status](#) endpoint to check the status.**

Deposits Using UPI

To make deposits from a user via UPI, initiate a deposit by transferring funds from the user's UPI account to your merchant's wallet using the [Transaction/UPIDeposit](#) endpoint. This method applies to INR currency.

Flow



Steps

- To initiate the UPI deposit transaction use [Transaction/UPIDeposit](#) endpoint.
- Display QR code (Received in response) the User will use this to do UPI transactions using the UPI application
- Wait for callback
- Check Status

Don't use your actual UPI application during the integration.

You can get the actual status [Transaction/CheckStatus](#) endpoint if you need this otherwise you don't need to call this endpoint, you will get a callback on your approval webhook URL once this transaction is created & approved by the user.

Once you've called the [Transaction/UPIDeposit](#) endpoint, a pending transaction is generated in the Xprizo system and sends a request Acquirer network to generate a QR code for the UPI payment. This QR code will come in response that will be displayed to the User to make a UPI deposit. Xprizo will continuously monitor this pending transaction for up to 30 minutes to get confirmation or failure for this transaction from the Acquirer's network. **Generally After 30 minutes, this transaction will automatically cancelled by the Acquirer if the user doesn't use the QR code to make a payment, but you should always check the transaction status before making any decision.**

- If a confirmation is received, the pending transaction will be marked as a completed transaction.
- If the transaction is canceled, a notification containing details about the cancellation will be generated and sent to the merchant's account.

Merchants will receive updates on the transaction status via callbacks. To enable callbacks, you'll need to configure this on your account (the profile used to create the transaction). You can accomplish this by navigating to the settings/preferences in Xprizo and configuring the "Approval Webhook" to point to a URL of your choice. Alternatively, you can achieve the same using the [Preference/SetApprovalWebhook](#) endpoint.

A callback will be sent when

- the pending transaction is created.
- the pending transaction is approved
- the pending transaction is cancelled

It is not strictly necessary to use callbacks, you have the option to periodically check the status of the transaction. However, it's not recommended as it would require to poll the server continuously to check the status. However, it's advisable to verify the status after receiving a callback to ensure that the transaction was completed successfully.

Use this endpoint to check the status [Transaction/Status](#).

**** You should, however, check the status, after you get a callback to confirm that the transaction was completed successfully. You can use [Transaction/Status](#) endpoint to check the status.**

Withdrawal

Withdrawal to Xprizo Wallets

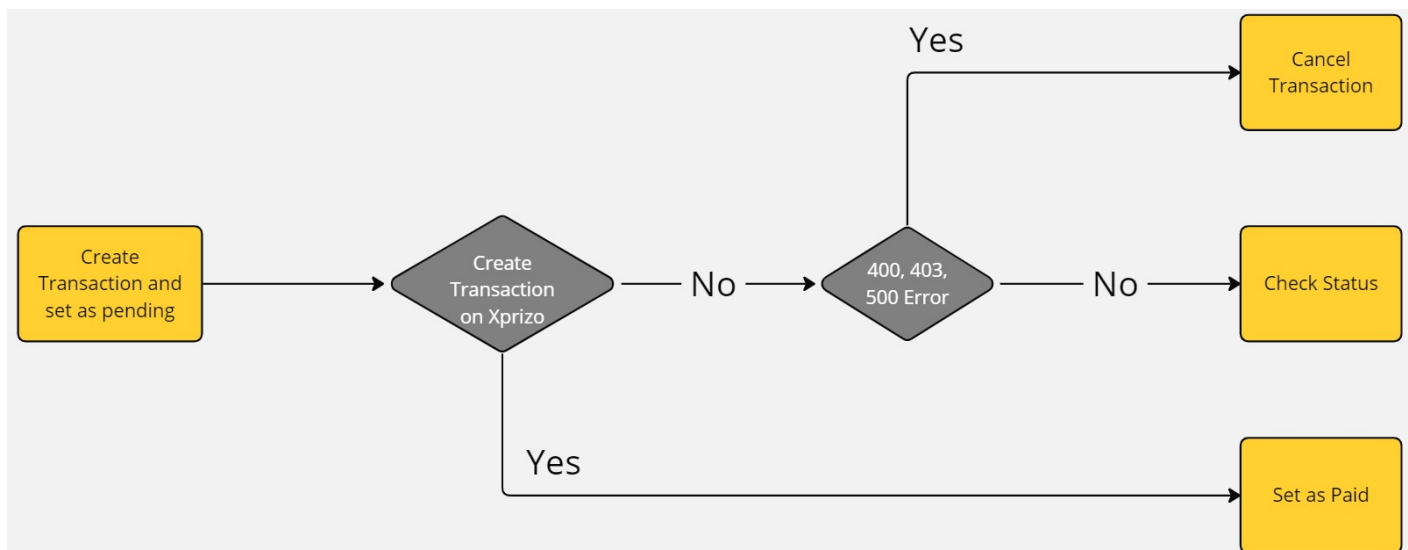
Withdrawals are carried out when a merchant wants to transfer funds from their wallet to a user's wallet. These transactions are initiated through the process of generating withdrawal requests. Typically, withdrawals can be done using three different methods. Each method provides merchants with flexibility in how they transfer funds, catering to their specific preferences and requirements. This flexibility ensures seamless fund transfers between merchants and users within the platform.

Create, Commit, Check

Employ this method if the final check is done on the platform's end, that is the platform verifies that the transaction was completed and will then update the user's wallet accordingly.

This method requires the platform to continually poll Xprizo in case there is no response to confirm whether the transaction was successfully created or not.

Flow



Steps

- Create a pending transaction on your system.
- Create and commit a payment transaction on Xprizo and wait for the response.

Transaction/SendPayment

- If you receive an Ok Response (200), then the transaction was successfully processed and you can set your payment transaction as completed.
- If you receive one of the following error codes (400,401,403,500), then the transaction did not complete and you should cancel your transaction (or correct the

problem and try again).

- If you do not receive a response from Xprizo or a timeout you will need to keep checking Xprizo until you can determine the status of the transaction.

Transaction/Status

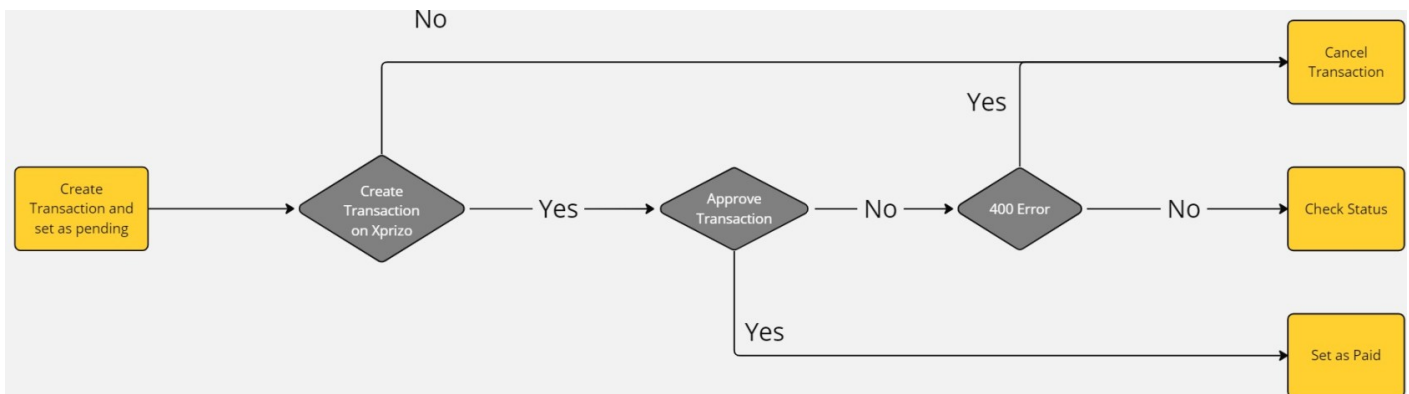
- Check transaction status
 - Active - Approved
 - Pending - Pending approval
 - NotFound - Timed-out or cancelled

Create, Approve, Check

Use this method if the creation and approval of transactions are not done at the same time. For instance, the platform might initiate the transaction, and the merchant may approve the transaction

With this method, the platform needs to continuously poll Xprizo in case there's no immediate response from Xprizo to confirm whether the transaction was successfully created or not.

Flow



Steps

- Create a pending transaction on your system.
- Create a payment transaction on Xprizo and wait for the response.

Transaction/SendPayment

- Use the key from the returned response to approve the transaction

Transaction/Approval or wait for the merchant to approve the transaction (you can only approve the transaction if the merchant gave you full or approval rights to their wallet).

- If you are approving the transaction
 - If you get an OK Response (200) then you should check the status of the transaction.

Transaction/Status

- If you will wait for the merchant to approve the transaction

- and you have set the approval callback option in your settings

Preference/SetApprovalWebhook then you will receive a callback once the transaction has been approved

- and you have **NOT** set the approval callback option then you will have to keep checking the status of the transaction until it is approved, rejected or expires.

Transaction/Status

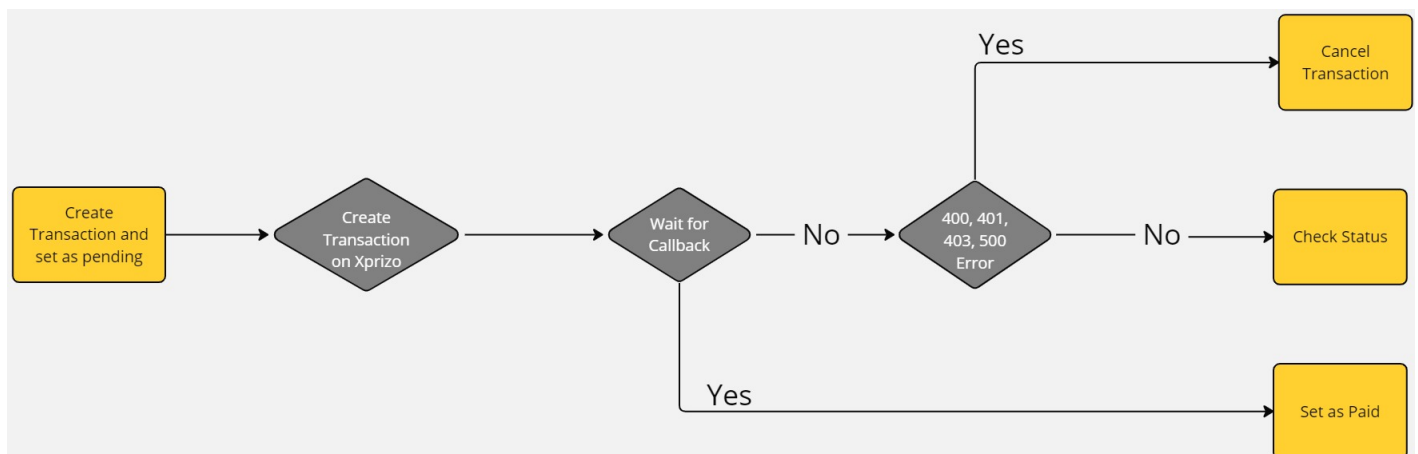
At any time you can cancel the transaction by calling [Transaction/Cancel](#)

Create, Hold, Check

Use this method if you prefer Xprizo to conduct the final check. Xprizo will verify that the transaction has been created on the platform by sending a callback and awaiting a valid response. Depending on the platform's response, Xprizo will either confirm the transaction or cancel it.

By using this method involves Xprizo initiating a transaction and placing it on hold. It will then send a callback to verify whether the platform has successfully updated the transaction on its end. If Xprizo receives a positive response, then the transaction will be completed. If a negative response is received or there is no response within 24 hours, the transaction is voided.

Flow



Steps

- Create a pending transaction on your system
- Create a payment transaction on Xprizo and wait for the response.

Transaction/SendPayment

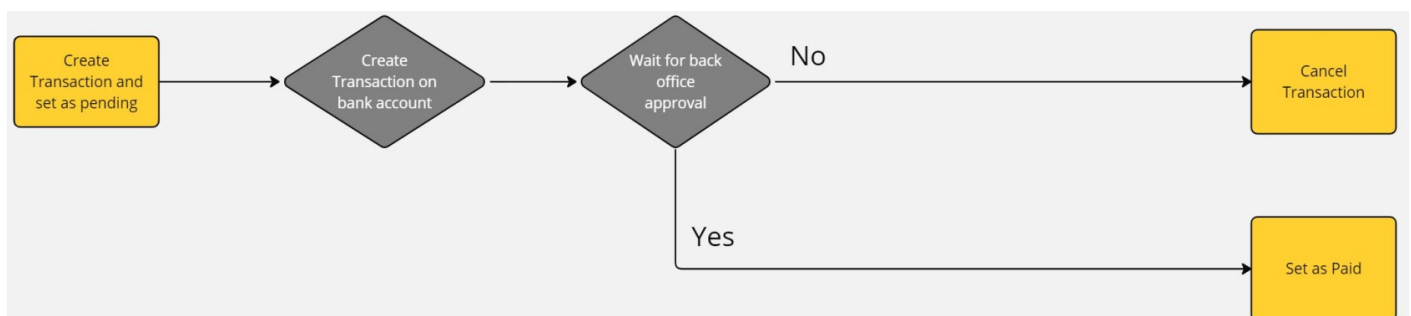
- Xprizo will create a transaction marked as 'Hold' and return 200 (success)
- If you get the 200 (success) response then leave your transaction as pending

- If the transaction failed with any error 400,401,403,500 excluding a timeout mark the transaction as rejected (Xprizo automatically deleted failed transactions)
- Xprizo will send a callback to the platform (make sure you have the "Payment Webhook" setup in Xprizo) **Preference/SetApprovalWebhook**.
 - You should now set the transaction on your platform as paid and respond to the callback with 200 (success) response.
 - It is recommended to check the status of the transaction on Xprizo to ensure that is completed **Transaction/Status**.
 - Xprizo will keep sending callbacks until the platform responds
 - If your platform does not respond for 24 hours
 - The transaction is voided
 - Funds released back to the merchant
 - If the platform responds with a success
 - The Hold is removed and the transaction is marked as active
 - If the platform responds with an error
 - The transaction is voided
 - Funds released back to the merchant

Withdrawal to Bank Accounts

When a user needs to transfer funds from their wallet to someone's bank account, they can easily initiate this process by providing the recipient's name, account number, and the desired amount to be transferred. This withdrawal method facilitates direct transfers to bank accounts, ensuring seamless transactions between users and their chosen recipients.

Flow



Steps

- Create a pending transaction on your system.
- Create a withdrawal request on Xprizo using this endpoint **Transaction/BankWithdrawal**
- After this the entered amount will be debited from your Xprizo wallet and the withdrawal will be created as a Hold transaction. This Hold transaction will be reviewed by the back

office for approval.

- If the back office user
 - approves of the transaction will go into the processing and after the successful processing this transaction will be marked as Active and the user will be notified using your approval webhook. You can configure your approval webhook using this [Preference/SetApprovalWebhook](#).
 - rejects the transaction the amount deducted will be reverted back to the user's wallet and the transaction will be marked as Void.
 - and you have **NOT** set the approval callback option then you will have to keep checking the status of the transaction until it is approved, rejected, or expires.

[Transaction/Status](#)

FAQs

1. How to Generate Payment Link

2. How to View Routing Codes

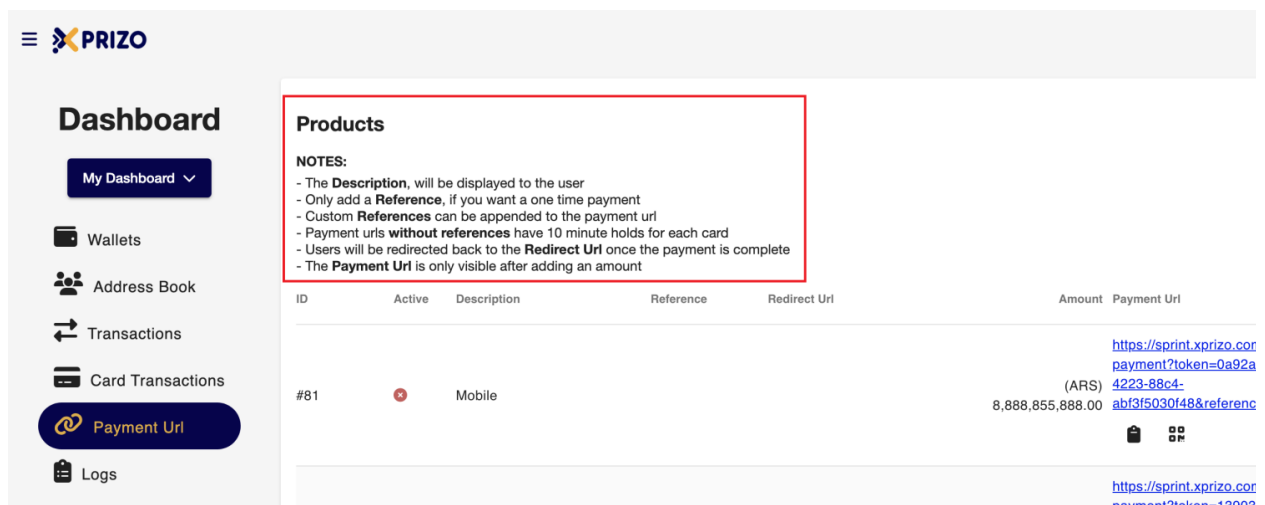
How to do Integration for Payment URL?

Introduction

The Payment Link feature in Xprizo allows users to create a shareable payment link by entering the product amount. This link can be shared with both Xprizo and non-Xprizo users, enabling them to make payments via various cards. Additionally, a QR code is generated along with the link for easy transactions. When the link is shared with another user, the specified amount is directly added to that user's wallet, streamlining the payment process and ensuring funds are quickly and securely transferred.

How to create Payment URL?

Follow the steps written in the image below to be able to create the Payment URL.



The screenshot shows the Xprizo dashboard with a sidebar on the left containing navigation links: Dashboard, My Dashboard, Wallets, Address Book, Transactions, Card Transactions, Payment Uri (highlighted), and Logs. The main content area is titled 'Products' and contains a 'NOTES' section with the following text:

- The **Description**, will be displayed to the user
- Only add a **Reference**, if you want a one time payment
- Custom **References** can be appended to the payment url
- Payment urls **without references** have 10 minute holds for each card
- Users will be redirected back to the **Redirect Uri** once the payment is complete
- The **Payment Uri** is only visible after adding an amount

Below the notes is a table with the following columns: ID, Active, Description, Reference, Redirect Uri, Amount, and Payment Uri. The table contains one row with the following data:

| ID | Active | Description | Reference | Redirect Uri | Amount | Payment Uri |
|-----|--------|-------------|-----------|--------------|------------------------|---|
| #81 | | Mobile | | | (ARS) 8,888,855,888.00 | https://sprint.xprizo.cor/payment?token=0a92a4223-88c4-abf3f5030f48&reference=13003 |

To receive the notifications for the payment you've received through webhook then you can set this by navigating to the settings of your profile page this process will be done from the UI side, For reference click on the link below.

[Create Payment URL Webhook](#)

To perform the same process with the help of API click on the below link for the reference.

[Approval Webhook](#)

