

Xprizo Manuals - API Manual

The API Endpoints Book is an essential guide for developers, providing comprehensive documentation on the various endpoints that constitute our application programming interface (API). This book serves as a roadmap for understanding, integrating, and optimizing interactions with our system. With a focus on clarity and practicality, developers can quickly find the information they need to seamlessly integrate our API into their applications, services, or projects.

- [Introduction](#)
- [Addressbook](#)
- [Transaction Endpoints \(api/Transaction\)](#)
- [Preference Endpoints \(api/Preference\)](#)
- [Wallet Endpoints \(api/Wallet\)](#)

Introduction

The purpose of this document is to provide a comprehensive guide to offer detailed insights into our API environments, and Authentication methods to be used at the time of integration.

1. Environment

1.1. Testing (Sandbox)

Before integrating with the live production system, you have the option to test features on our dedicated test server. Access sample data and simulated transactions to ensure a smooth and error-free integration process. Experiment with the functionalities in a controlled environment before deploying to the production system for a seamless experience.

Base URL for Sandbox <https://test.xprizo.com/api>

1.2 Production (Live)

Once you have completed your testing phase, you can transition to the live environment by directing your system to the production server.

Base URL for Live <https://wallet.xprizo.com/api>

2. Authentication

Xprizo employs JWT tokens and API keys as authentication methods for accessing API functions. You have the flexibility to choose either authentication method when interacting with the endpoints. You need to pass a JWT token or API key in the header of each request.

**** You need to use a processing profile to create a token or key to use all API endpoints.**

2.1. JSON Web Tokens (JWT)

2.1.1. Creating a Token

Creating a token is done by calling the GetToken endpoint with your username and password. If you are not a registered user you can register by going to the [Sandbox](#) or [Live](#) server.

Log in to the system using a username and password. You will receive back a JWT Bearer token that you will need to make secure API requests.

Request:

```
curl --location '{baseurl}/Security/GetToken' \  
--header 'accept: text/plain; x-api-version=1.0' \  
--header 'x-api-version: 1.0' \  
--header 'Content-Type: application/json; x-api-version=1.0' \  
--data-raw '{  
  "userName": "use username, mobile number to email",  
  "password": "your password"  
}'
```

Response:

```
{  
  "id": <a unique Id>,  
  "token": <use this token for future calls>,  
  "expires": <the time the token expires>  
}
```

2.1.2. Usages of Token

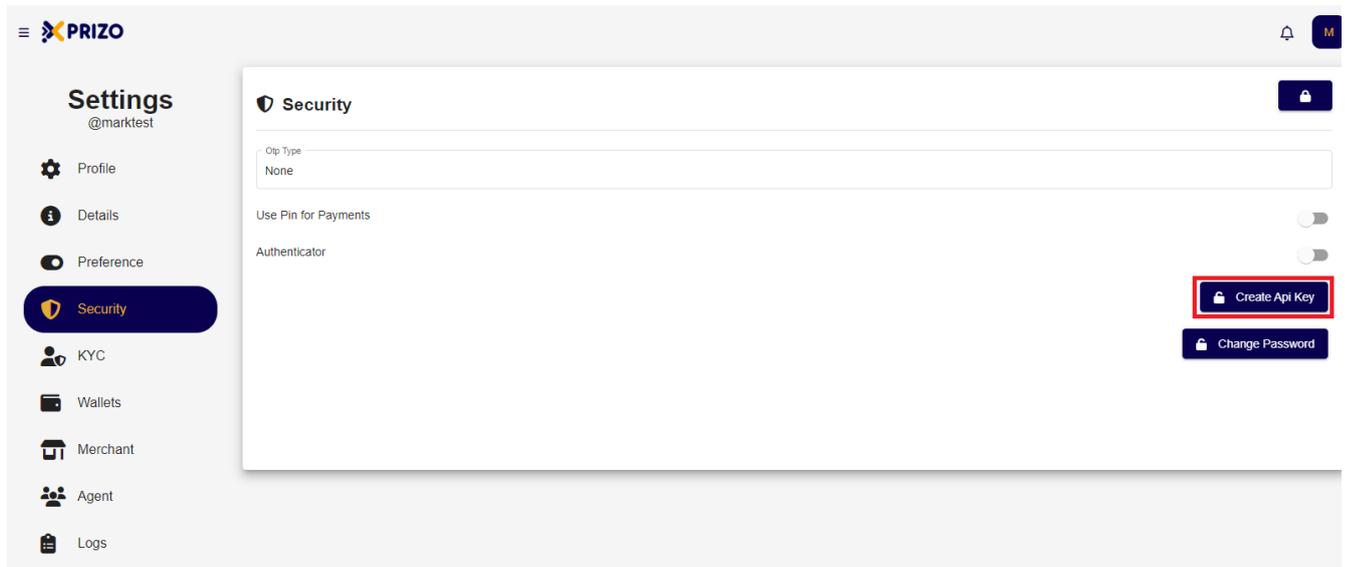
Use this token in the header of all secure requests to other endpoint

```
curl -X 'GET' \  
  '{baseurl}/Wallet/Info?contact=1' \  
-H 'accept: text/plain' \  
-H 'Authorization: bearer <token>
```

2.2. API Keys

2.2.1. Creating an API Key

To configure your API key, access your User Account, navigate to 'settings' by clicking the profile icon, and locate 'security' in the list. Within the security settings, you will find an option to set the API key. It's important to note that if the API key is removed, a new key must be issued and utilized for transactions.



2.2.2. Usages of API Key

Use the API key in the header in the following way.

```
curl --location --request GET
' {baseurl}/Wallet/Info?contact={username/emailid/mobilenumber}&currencyCode={currencyCode}' \
--header 'Accept: text/plain; x-api-version=1.0' \
--header 'x-api-version: {apiversion}' \
--header 'x-api-key: {apikey}'
```

2.3. Know your account ID

To submit deposits for processing, you will need your accountId (wallet ID) from your account. You can find this by the following process, login to your account and open settings, find 'wallet' on the list and you will find an ID associated with the wallet of required currency. If a wallet doesn't exist, you will be required to create one by clicking the  button.

Addressbook

Get AddressBook Item Information

Introduction

The `GET /api/Addressbook/Info` endpoint allows you to retrieve information about a specific item in the address book. You can search using a username, email, or mobile number.

Create Request

```
curl -X 'GET' \  
  'https://wallet.xprizo.com/api/Addressbook/Info?value=John%20Doe' \  
  -H 'accept: text/plain' \  
  -H 'x-api-version: 1'
```

Endpoint: Get Information about an AddressBook Item

- **URL:** `/api/Addressbook/Info`
- **Method:** `GET`
- **Description:** Retrieve information about an AddressBook item by providing a username, email, or mobile number.

Parameter	Description
Query	(string, required): Enter the username, email, or mobile number of the contact you are searching for.
Header	(string, required): Specify the API version.

Responses

Response (200 Success)

The request was successful, and the information about the AddressBook item is returned.

```
{
  "id": 0,
  "description": "string"
}
```

Errors

- **400 Bad Request** The request could not be understood or was missing required parameters.
- **401 Unauthorized** Authentication failed or the user does not have permission for the requested operation.
- **500 Server Error** An error occurred on the server.

Get Active User Count in Address Book

Introduction

The `GET /api/Addressbook/Count` endpoint allows you to retrieve the number of active users in the address book.

Create Request

```
curl -X 'GET' \
  'https://wallet.xprizo.com/api/Addressbook/Count?contactId=2131' \
  -H 'accept: text/plain' \
  -H 'x-api-version: 1'
```

Endpoint: Get Contact Count

- **URL:** `/api/Addressbook/Count`
- **Method:** `GET`
- **Description:** Retrieve the count of AddressBook contacts based on a specific contact ID.

Parameter	Description
Query	contactId (integer, required): The ID of the contact for which you want to get the count.

Header	x-api-version (string, required): Specify the API version.
--------	--

Responses

Response (200 Success)

The request was successful, and the information about the AddressBook item is returned.

Errors

- **400 Bad Request** The request could not be understood or was missing required parameters.
- **401 Unauthorized** Authentication failed or the user does not have permission for the requested operation.
- **500 Server Error** An error occurred on the server.

Set User KYC Compliant

Introduction

The `SetUserKYCCompliant` API endpoint allows you to update the KYC (Know Your Customer) compliance status of a user in the address book. This endpoint requires proper authentication and versioning headers.

Create Request

```
curl -X 'PUT' \  
  'https://wallet.xprizo.com/api/Addressbook/SetUserKYCCompliant/123?value=true' \  
  -H 'accept: */*' \  
  -H 'x-api-version: 1'
```

Endpoint: Set User KYC Compliant

- **URL:** `/api/Addressbook/SetUserKYCCompliant/{contactId}`
- **Method:** `PUT`
- **Description:** Update the KYC compliance status of a specific user in the address book.

Parameter	Description
-----------	-------------

Path	contactId (integer, required): The ID of the contact whose KYC compliance status is being updated.
Query	contactId (integer, required): The ID of the contact for which you want to get the count.
Header	x-api-version (string, required): Specify the API version.

Responses

Response (200 Success)

The request was successful, and the information about the AddressBook item is returned.

Errors

- **400 Bad Request** The request could not be understood or was missing required parameters.
- **401 Unauthorized** Authentication failed or the user does not have permission for the requested operation.
- **500 Server Error** An error occurred on the server.

Rename Contact in Address Book

Introduction

The `PUT /api/Addressbook/RenameUser` endpoint allows you to rename a contact in the address book.

Create Request

```
curl -X 'PUT' \
  'https://wallet.xprizo.com/api/Addressbook/RenameUser' \
  -H 'accept: */*' \
  -H 'x-api-version: 1' \
  -H 'Content-Type: application/json' \
  -d '{
  "id": 0,
```

```
"description": "string"
}'
```

Endpoint: Rename Contact

- **URL:** `/api/Addressbook/RenameUser`
- **Method:** `PUT`
- **Description:** Rename a contact in the address book.

Parameter	Description
Header	x-api-version (string, required): Specify the API version.

Responses

Response (200 Success)

The request was successful, and the information about the AddressBook item is returned.

Errors

- **400 Bad Request** The request could not be understood or was missing required parameters.
- **401 Unauthorized** Authentication failed or the user does not have permission for the requested operation.
- **500 Server Error** An error occurred on the server.

Disable Contact in Address Book

Introduction

The `DELETE /api/Addressbook/Disable/{contactId}` endpoint allows you to disable a contact in the address book.

Create Request

```
curl -X 'DELETE' \  
  'https://wallet.xprizo.com/api/Addressbook/Disable/123' \  
  -H 'accept: */*' \  
  -H 'x-api-version: 1'
```

Endpoint: Disable Contact

- **URL:** `/api/Addressbook/Disable/{contactId}`
- **Method:** `DELETE`
- **Description:** Disable a contact in the address book.

Parameter	Description
Path	contactId (integer, required): The ID of the contact whose KYC compliance status is being updated.
Header	x-api-version (string, required): Specify the API version.

Responses

Response (200 Success)

The request was successful, and the information about the AddressBook item is returned.

Errors

- **400 Bad Request** The request could not be understood or was missing required parameters.
- **401 Unauthorized** Authentication failed or the user does not have permission for the requested operation.
- **500 Server Error** An error occurred on the server.

Enable Contact in Address Book

Introduction

The `DELETE /api/Addressbook/Enable/{contactId}` endpoint allows you to enable a previously disabled contact in the address book.

Create Request

```
curl -X 'DELETE' \
  'https://wallet.xprizo.com/api/Addressbook/Enable/123' \
  -H 'accept: */*' \
  -H 'x-api-version: 1'
```

Endpoint: Enable Contact

- **URL:** `/api/Addressbook/Enable/{contactId}`
- **Method:** `DELETE`
- **Description:** Enable a previously disabled contact in the address book.

Parameter	Description
Path	contactId (integer, required): The ID of the contact whose KYC compliance status is being updated.
Header	x-api-version (string, required): Specify the API version.

Responses

Response (200 Success)

The request was successful, and the information about the AddressBook item is returned.

Errors

- **400 Bad Request** The request could not be understood or was missing required parameters.
- **401 Unauthorized** Authentication failed or the user does not have permission for the requested operation.
- **500 Server Error** An error occurred on the server.

Add Contact to Address Book

Introduction

The `POST /api/Addressbook/Add` endpoint allows you to add a new contact to the address book. The contact information can be a phone number, email, or a name.

Create Request

```
curl -X 'POST' \  
  'https://wallet.xprizo.com/api/Addressbook/Add' \  
  -H 'accept: */*' \  
  -H 'x-api-version: 1' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "value": "string"  
  }'
```

Endpoint: Add Contact

- **URL:** `/api/Addressbook/Add`
- **Method:** `POST`
- **Description:** Add a new contact to the address book.

Parameter	Description
Header	x-api-version (string, required): Specify the API version.

Responses

Response (200 Success)

The request was successful, and the information about the AddressBook item is returned.

Errors

- **400 Bad Request** The request could not be understood or was missing required parameters.
- **401 Unauthorized** Authentication failed or the user does not have permission for the requested operation.
- **500 Server Error** An error occurred on the server.

Validate a User Using Rules

Introduction

The `POST /api/Addressbook/ValidateUser` endpoint allows you to validate a user based on specific rules. The validation can be performed using a contact name, which can be a phone number, an email address, or a name.

Create Request

```
curl -X 'POST' \
  'https://wallet.xprizo.com/api/Addressbook/ValidateUser?ruleName=test' \
  -H 'accept: text/plain' \
  -H 'x-api-version: 1' \
  -H 'Content-Type: application/json' \
  -d '{
  "value": "string"
}'
```

Endpoint: Validate User

- **URL:** `/api/Addressbook/ValidateUser`
- **Method:** `POST`
- **Description:** Validate a user using specified rules.

Endpoint: Set User KYC Compliant

- **URL:** `/api/Addressbook/SetUserKYCCompliant/{contactId}`
- **Method:** `PUT`
- **Description:** Update the KYC compliance status of a specific user in the address book.

Parameter	Description
Request	value (string, required): The contact information to be validated. This can be a phone number, email, or name.
Query	contactId (integer, required): The ID of the contact for which you want to get the count.
Header	x-api-version (string, required): Specify the API version.

Responses

Response (200 Success)

The request was successful, and the information about the AddressBook item is returned.

```
{
  "id": 0,
  "description": "string"
}
```

Errors

- **400 Bad Request** The request could not be understood or was missing required parameters.
- **401 Unauthorized** Authentication failed or the user does not have permission for the requested operation.
- **500 Server Error** An error occurred on the server.

Delete an Address Book Contact

Introduction

The `DELETE /api/Addressbook/Delete/{contactId}` endpoint allows you to delete a specific contact from the address book.

Create Request

```
curl -X 'DELETE' \
  'https://wallet.xprizo.com/api/Addressbook/Delete/123' \
  -H 'accept: */*' \
  -H 'x-api-version: 1'
```

Endpoint: Delete Address Book Contact

- **URL:** `/api/Addressbook/Delete/{contactId}`
- **Method:** `DELETE`
- **Description:** Delete a specific contact from the address book.

Parameter	Description
Path	contactId (integer, required): The unique ID of the contact to be deleted.
Header	x-api-version (string, required): Specify the API version.

Responses

- **200 Success** The request was successful, and the information about the AddressBook item is returned.
- **400 Bad Request** The request could not be understood or was missing required parameters.
- **401 Unauthorized** Authentication failed or the user does not have permission for the requested operation.
- **500 Server Error** An error occurred on the server.

Transaction Endpoints (api/Transaction)

General Definitions

Parameter	Description
baseurl	Sandbox - https://test.xprizo.com/api Live- https://wallet.xprizo.com/api
apiversion	1.0
apikey	API key For authentication

To obtain the API key for authentication, click [here](#) to generate the API key.

Create Deposit Request Using a Card

Request

```
curl --location '{baseurl}/Transaction/CardDeposit' \  
--header 'Accept: text/plain; x-api-version=1.0' \  
--header 'Content-Type: application/json; x-api-version=1.0' \  
--header 'x-api-version: {apiversion}' \  
--header 'x-api-key: {apikey}' \  
--data '{  
  "description": <purpose of deposit>,  
  "reference": <unique reference>,  
  "amount": <the deposit amount>,  
  "accountId": <merchants wallet id>,  
  "transferAccountId": <wallet id where deposit amount will be transferred>,  
  "customer": <unique name for the depositor>,  
  "creditCard": {
```

```

"name": <name on card>,
"number": <card number>,
"expiryMonth":<2 diget expiry month 01-12>,
"expiryYear": <2 diget expiry year>,
"cvv": <3 or 4 digets>
},
"routingCode": "string",
"redirect": "string"
}'

```

Parameter	Description
accountId	(Required) This is the ID of the recipient's wallet - the wallet into which the funds will be deposited. You can check your account id here
transferAccountId	(Optional) This is the ID of the recipient's wallet - the wallet where the funds should be transferred after a deposit has been made to the main wallet. **Note: This is required when you want to deposit money in a wallet of currency that is not supported by the acquirer. For this, we will do an internal transfer and at the time of the transfer currency conversion charges will be applicable.
reference	(Required) The reference is the unique transaction identifier for this transaction. This will be generated by processor user and will be used in status checks and callbacks.
customer	(Required) A unique identifier of the person making the deposit. If this customer is not found in the system, they will be added to recipient's the address book. You can pass email id of that user.
amount	(Required) The deposit amount, with 2 decimal places, using dot as the decimal.
description	(Optional) Text that can be used to describe the reason for the deposit. In Test Mode. use:- <ul style="list-style-type: none"> • Pass, Success - the transaction will be created as approved. • Reject, Fail, Cancel - the transaction will be created and then rejected.
routingCode	Use one of the routing codes that have been set on your account (routing codes are used for different currencies and limits). This will be configured by the Xprizo Team and you can find these here.
redirect	(Required for 3DS) If the card is processed using 3ds then you will need to open a confirmation screen (the redirect URL for the confirmation screen is returned in the response to this request). This redirect is the URL that the user will be returned to after the confirmation screen has been closed.

Response (200)

```
{
  "key": <a unique identifier for the transaction>,
  "statusCode": <The number equivalent of the Status>,
  "status": <Active/Rejected/Pending/Redirect>,
  "value": <billingcode/reason/url>
}
```

Parameter	Description
key	a unique identifier for the transaction
statusCode	The number equivalent of the Status
status	<ul style="list-style-type: none">• Active - The transaction was completed successfully• Pending - The transaction was unable to be completed and needs further action• Rejected - The transaction was rejected• Redirect - (3DS) the user needs to be redirected to another URL to complete the transaction
value	<ul style="list-style-type: none">• Active - the value will show the description• Pending - the value will show the description• Rejected - the value will show the reason• Redirect - the value will show the URL to redirect to

Errors

- **400** (Bad Request) The transaction could not be completed. The reason is returned Check the error, contact Xprizo, or fix the problem and try again.

Message	Detail of Message
Invalid routing code	Your account does not have the routing code configured. Please contact the Xprizo team to have this set up.
Invalid routing code (No Mid Set)	Your account does not have the MID configured. Please reach out to the Xprizo team to have this configured.

A transaction with this reference already exists	You need to generate a new reference and try again
A pending transaction with this reference already exists (Ref:652-1706532591287))	The transaction is already available for this reference number with status pending, please check the status of this or try again with a new reference number

- **401** (Unauthorized) Invalid or expired token.
- **403** (Forbidden) You do not have the right to create or approve this transaction.
- **500** (Internal Server Error) A critical unexpected system error has occurred. Contact Xprizo and report the error.

Test Configuration For Routing Options

Routing Option	Routing Code Prefix	Configuration
Option A	MAN	<p>Amount</p> <ul style="list-style-type: none"> • \$1.00 or \$2.00 - Active(2DS Success) • \$3.00 or \$4.00 - Rejected • \$5.00 or \$6.00 - Rdirect (3DS Redirect URL) • \$7.00 or greater - Rejected Case In 3DS

Create a Deposit Request Using the Xprizo Wallet

Request

```
curl --location '{baseurl}/Transaction/RequestPayment' \
--header 'Accept: text/plain; x-api-version=1.0' \
--header 'Content-Type: application/json; x-api-version=1.0' \
--header 'x-api-version: {apiversion}' \
--header 'x-api-key: {apikey}' \
```

```
--data '{
  "fromAccountId": <users_account_id>,
  "toAccountId": <your_account_id>,
  "description": <purpose of deposit>,
  "amount": <requested amount>,
  "reference": <unique reference>
}'
```

Parameter	Description
fromAccountId	This is the ID of the payee's wallet - the user who is paying out the funds
toAccountId	This is the ID of recipient's wallet - the user who will receive the funds
description	(Optional) Text that can be used to describe the reason for the transaction
amount	The amount of the transaction
reference	A unique reference is created so that you can identify and link this transaction to your system

Response (200)

```
{
  "key": <a unique identifier for the transaction>,
  "approveById": <id of the person to approve the transaction>,
  "canCancel": <can be cancelled or not>,
  "ttl": <time left in seconds>,
  "expiryDate": <date/time when the transaction expires>,
  "error": ""
}
```

Parameter	Description
key	a unique identifier for the transaction
approveById	<id of the person to approve the transaction>
canCancel	<can be cancelled or not>
expiryDate	<date/time when the transaction expires>
ttl	<time left in seconds>

Errors

- **400** (Bad Request) The transaction could not be completed. The reason is returned Check the error, contact Xprizo, or fix the problem and try again.
- **401** (Unauthorized) Invalid or expired token.
- **403** (Forbidden) You do not have the right to create or approve this transaction.
- **500** (Internal Server Error) A critical unexpected system error has occurred. Contact Xprizo and report the error.

Create a Deposit Request Using Mpesa Wallet

Request

```
curl --location '{baseurl}/Transaction/MPesaDeposit' \
--header 'Accept: text/plain; x-api-version=1.0' \
--header 'Content-Type: application/json; x-api-version=1.0' \
--header 'x-api-version: {apiversion}' \
--header 'x-api-key: {apikey}' \
--data '{
  "mobileNumber": <mpesa_mobile_number>,
  "accountId": <your_account_id>,
  "description": <purpose of deposit>,
  "amount": <requested amount>,
  "reference": <unique reference>
}'
```

Parameter	Description
mobileNumber	Mpesa Mobile number to be used for transaction

accountId	This is the ID of recipient's wallet - the user who will receive the funds. You can check your account id here
description	(Optional) Text that can be used to describe the reason for the transaction. In Test Mode. use:- <ul style="list-style-type: none"> • Pass, Success - the transaction will be created as approved. • Reject, Fail, Cancel - the transaction will be created and then rejected.
amount	The amount of the transaction
reference	A unique reference is created so that you can identify and link this transaction to your system

Response (200)

```
{
  "key": <a unique identifier for the transaction>,
  "statusCode": <The number equivalent of the Status>,
  "status": <Active/Rejected/Pending>,
  "value": <transaction description>
}
```

Parameter	Description
key	a unique identifier for the transaction
statusCode	The number equivalent of the Status
status	<ul style="list-style-type: none"> • Active - The transaction was completed successfully • Pending - The transaction was unable to be completed and needs further action • Rejected - The transaction was rejected
value	<ul style="list-style-type: none"> • Active - the value will show the description • Pending - the value will show the description • Rejected - the value will show the reason

Errors

- **400** (Bad Request) The transaction could not be completed. The reason is returned Check the error, contact Xprizo, or fix the problem and try again.

- **401** (Unauthorized) Invalid or expired token.
- **403** (Forbidden) You do not have the right to create or approve this transaction.
- **500** (Internal Server Error) A critical unexpected system error has occurred. Contact Xprizo and report the error.

Create a Withdrawal Request Using Mpesa

Request

```
curl --location '{baseurl}/Transaction/MPesaWithdrawal' \
--header 'Accept: text/plain; x-api-version=1.0' \
--header 'Content-Type: application/json; x-api-version=1.0' \
--header 'x-api-version: {apiversion}' \
--header 'x-api-key: {apikey}' \
--data '{
  "mobileNumber": <mpesa_mobile_number>,
  "accountId": <your_account_id>,
  "description": <purpose of deposit>,
  "amount": <requested amount>,
  "reference": <unique reference>
}'
```

Parameter	Description
mobileNumber	Mpesa Mobile number to be used for transactions. Who will receive the funds. Sample +254706****73
accountId	This is the ID of your wallet - You can check your account ID here
description	(Optional) Text that can be used to describe the reason for the transaction. In Test Mode. use:- <ul style="list-style-type: none"> • Pass, Success - the transaction will be created as approved. • Reject, Fail, Cancel - the transaction will be created and then rejected.
amount	The amount of the transaction

reference	A unique reference is created so that you can identify and link this transaction to your system
-----------	---

Response (200)

```
{
  "key": <a unique identifier for the transaction>,
  "statusCode": <The number equivalent of the Status>,
  "status": <Active/Rejected/Pending>,
  "value": <transaction description>
}
```

Parameter	Description
key	a unique identifier for the transaction
statusCode	The number equivalent of the Status
status	<ul style="list-style-type: none"> Active - The transaction was completed successfully Pending - The transaction was unable to be completed and needs further action Rejected - The transaction was rejected
value	<ul style="list-style-type: none"> Active - the value will show the description Pending - the value will show the description Rejected - the value will show the reason

Errors

- **400** (Bad Request) The transaction could not be completed. The reason is returned Check the error, contact Xprizo, or fix the problem and try again.
- **401** (Unauthorized) Invalid or expired token.
- **403** (Forbidden) You do not have the right to create or approve this transaction.
- **500** (Internal Server Error) A critical unexpected system error has occurred. Contact Xprizo and report the error.

Create a Deposit Request Using UPI



Request

```
curl --location '{baseurl}/Transaction/UpiDeposit' \  
--header 'Accept: text/plain; x-api-version=1.0' \  
--header 'Content-Type: application/json; x-api-version=1.0' \  
--header 'x-api-version: {apiversion}' \  
--header 'x-api-key: {apikey}' \  
--data '{  
  "accountId": <your_account_id>,  
  "description": <purpose of deposit>,  
  "amount": <requested amount>,  
  "reference": <unique reference>  
'
```

Parameter	Description
accountId	This is the ID of the recipient's wallet - the user who will receive the funds. You can check your account ID here
description	(Optional) Text that can be used to describe the reason for the transaction.
amount	The amount of the transaction
reference	A unique reference is created so that you can identify and link this transaction to your system

Response (200)

```
{  
  "key": <a unique identifier for the transaction>,  
  "statusCode": <The number equivalent of the Status>,  
  "status": <status>,  
  "description": <Content of QR Code>  
  "value": <Image date of QR code>  
}
```

Parameter	Description
key	a unique identifier for the transaction
statusCode	The number equivalent of the Status

status	Pending - The transaction request is created and needs further action by the user. In Test Mode, You need to simulate the success case using the below-mentioned endpoint
value	Image data for QR code, you need to use this to display on User Interface.
description	Content of QR code

When utilizing the UpiDeposit function, do not send anything in the redirect field to opt for the QrCode option. Omitting the redirect triggers the IFrame screen, but without it, you'll receive a response containing the UPI QrCode. Display this QrCode to the user for seamless integration.

Errors

- **400** (Bad Request) The transaction could not be completed. The reason is returned Check the error, contact Xprizo, or fix the problem and try again.
- **401** (Unauthorized) Invalid or expired token.
- **403** (Forbidden) You do not have the right to create or approve this transaction.
- **500** (Internal Server Error) A critical unexpected system error has occurred. Contact Xprizo and report the error.

Validate UPI Deposit

The UpiDepositResponse function is utilized by our Acquirer. They invoke this function once the user has processed the transaction or after approximately 30 minutes to cancel the QrCode if it remains unused.

Upon calling the UpiDeposit endpoint and receiving the QrCode, a new transaction is presented in the wallet with a "Pending" status. This status persists until our acquirer either validates or cancels the QrCode using the UpiDepositResponse function.

During testing, an option is provided to call this function by passing the Key of the request generated using UpiDeposit. In test mode, invoking this function automatically "Approves" the transaction, transitioning it to an active state.

Request

```
curl --location '{baseurl}/Transaction/UpiDepositResponse?key={key}' \  
--header 'Accept: text/plain; x-api-version=1.0' \  
--header 'x-api-version: {apiversion}' \  

```

```
--header 'x-api-key: {apikey}'
```

Parameter	Description
key	a unique identifier for the transaction

Response (200)

```
{  
  "key": <a unique identifier for the transaction>,  
  "statusCode": <The number equivalent of the Status>,  
  "status": <Active/Pending/Cancelled>  
}
```

Parameter	Description
key	a unique identifier for the transaction
statusCode	The number equivalent of the Status
status	Active/Pending/Cancelled

Errors

- **400** (Bad Request) The transaction could not be completed. The reason is returned Check the error, contact Xprizo, or fix the problem and try again.
- **401** (Unauthorized) Invalid or expired token.
- **403** (Forbidden) You do not have the right to create or approve this transaction.
- **500** (Internal Server Error) A critical unexpected system error has occurred. Contact Xprizo and report the error.

Send Payment Using Xprizo Wallet

Request

```
curl --location '{baseurl}/Transaction/SendPayment?action=<action>&pin=<pin>' \  
--header 'Accept: text/plain; x-api-version=1.0' \  

```

```
--header 'Content-Type: application/json; x-api-version=1.0' \
--header 'x-api-version: {apiversion}' \
--header 'x-api-key: {apikey}' \
--data '{
  "fromAccountId": <your_account_id>,
  "toAccountId": <users_account_id>,
  "reference": <unique reference>,
  "amount": <requested amount>,
  "description": <purpose of payment>
}'
```

Parameter	Description
action	0 or default (Transaction will go in approval mode and payee needs to approve the transaction) 1 (Transaction will be completed with any approval) 2 (Transaction will be on hold until Xprizo will receive HTTP code 200 on payment webhook)
pin	Required if the user implemented "Approval Security"
fromAccountId	This is the ID of the payee's wallet - the user who is paying out the funds
toAccountId	This is the ID of the recipient's wallet - the user who will receive the funds
description	(Optional) Text that can be used to describe the reason for the transaction
amount	The amount of the transaction
reference	A unique reference is created so that you can identify and link this transaction to your system

Response (200) If action = 0 or Not provided in request

```
{
  "key": <a unique identifier for the transaction>,
  "approveById": <id of the person to approve the transaction>,
  "canCancel": <can be cancelled or not>,
  "ttl": <time left in seconds>,
```

```
"expiryDate": <date/time when the transaction expires>,  
"error": ""  
}
```

Parameter	Description
key	a unique identifier for the transaction
approveById	<id of the person to approve the transaction>
canCancel	<can be cancelled or not>
expiryDate	<date/time when the transaction expires>
ttl	<time left in seconds>

Response (200) If action = 1 or 2

```
{  
  "id": <a unique identifier for the transaction>,  
  "description": "<The type of transaction processed (Payment)>"  
}
```

Errors

- **400** (Bad Request) The transaction could not be completed. The reason is returned Check the error, contact Xprizo, or fix the problem and try again.
- **401** (Unauthorized) Invalid or expired token.
- **403** (Forbidden) You do not have the right to create or approve this transaction.
- **500** (Internal Server Error) A critical unexpected system error has occurred. Contact Xprizo and report the error.

Approve Payment Created Using Xprizo Wallet

Request

```
curl --location '{baseurl}/Transaction/ApprovalAccept?key=<key>&pin=<pin>&once=<true/false>' \  
--header 'Accept: text/plain; x-api-version=1.0' \  

```

```
--header 'Content-Type: application/json; x-api-version=1.0' \  
--header 'x-api-version: {apiversion}' \  
--header 'x-api-key: {apikey}'
```

Parameter	Description
key	A unique ID of the transaction you wish to approve
pin	Required if the user implemented "Approval Security"
once	Set this to true, if you want to automatically cancel this transaction if an error occurs when processing the transaction.

Response (200)

```
{  
  "id": <a unique identifier for the transaction>,  
  "description": "<The type of transaction processed (Payment)>"  
}
```

Cancel Payment Created Using Xprizo Wallet

Request

```
curl --location '{baseurl}/Transaction/ApprovalCancel?key=<key>' \  
--header 'Accept: text/plain; x-api-version=1.0' \  
--header 'Content-Type: application/json; x-api-version=1.0' \  
--header 'x-api-version: {apiversion}' \  
--header 'x-api-key: {apikey}'
```

Parameter	Description
key	A unique ID of the transaction you wish to approve

Response (200) Transaction is canceled successfully

Transaction Status Check

This endpoint can be used to check the status of any type of transaction for example card deposit, Mpesa withdrawal etc.

Request

```
curl --location '{baseurl}/Transaction/Status/{accountId}?reference={reference}' \
--header 'x-api-version: {apiversion}' \
--header 'x-api-key: {apikey}'
```

Parameter	Description
accountId	This is the ID of the wallet that contains the transaction. When requesting a payment use the ID of the recipient's wallet.
reference	The reference is the Merchants unique transaction identifier for this transaction

Response (200)

```
{
  "key" : <a unique identifier for the transaction>
  "statusCode" : <0/3/1/-1/4/5/6>
  "status" : <Active/Rejected/Pending/NotFound/Cancelled/Hold/Void >
  "value" : <billingdescription/transaction description/Void reason>
}
```

Pending Transaction

```
{
  "key": "",
  "statusCode": 1,
  "status": "Pending",
  "value": "Pass"
```

```
}
```

```
# Approved Transaction
```

```
{
```

```
  "key": "3169",
```

```
  "statusCode": 0,
```

```
  "status": "Active",
```

```
  "value": "testing.com Pass"
```

```
}
```

```
# Rejected Transaction
```

```
{
```

```
  "key": "144",
```

```
  "statusCode": 3,
```

```
  "status": "Rejected",
```

```
  "value": "Failed Test"
```

```
}
```

```
# Not Found Transaction
```

```
{
```

```
  "key": "",
```

```
  "statusCode": -1,
```

```
  "status": "NotFound",
```

```
  "value": ""
```

```
}
```

```
# Hold Transaction (Mpesa Withdrawal)
```

```
{
```

```
  "key": "144",
```

```
  "statusCode": 5,
```

```
  "status": "Hold",
```

```
  "value": "Mpesa Withdrawal +254342222222"
```

```
}
```

```
# Void Transaction (Mpesa Withdrawal)
```

```

{
  "key": "144",
  "statusCode": 6,
  "status": "Void",
  "value": "342222222 Voided: No Response Data"
}

```

Parameter	Description
key	<p>This is the unique ID of the transaction.</p> <ul style="list-style-type: none"> • When Active or Hold or Void this will contain the transaction ID • When Rejected or Cancelled, it will be the ID given in the log file for the entry. • When pending or NotFound, it will be blank.
status	<p>The status of the transaction - It can be one of the following:</p> <ul style="list-style-type: none"> • Active - The transaction is a valid active transaction • Pending - The transaction is awaiting approval (or rejection). • NotFound - A transaction on this wallet, with this reference does not exist. • Rejected - the transaction has been rejected by the acquirer. • Cancelled - If Xprizo cancels the transaction due to technical error or invalid calls. • Hold - This is applicable for Mpesa withdrawal
value	<ul style="list-style-type: none"> • When Active this will contain the billing description For card deposit & transaction description for other transactions. • When Hold this will contain the withdrawal description • When Void this will contain the void reason • When Rejected or Cancelled, it will contain the reason. • When pending or NotFound, it will be blank

Preference Endpoints (api/Preference)

General Definitions

Parameter	Description
baseurl	Sandbox - https://test.xprizo.com/api Live- https://wallet.xprizo.com/api
apiversion	1.0
apikey	API key For authentication

Update Approval Webhook

Approval Webhook is triggered when a new pending transaction is created or when this pending transaction is approved, canceled, or rejected.

Request

```
curl --location --request PUT '{baseurl}/Preference/SetApprovalWebhook?url={url}' \  
--header 'Accept: text/plain; x-api-version=1.0' \  
--header 'x-api-version: {apiversion}' \  
--header 'x-api-key: {apikey}'
```

Parameter	Description
url	Url to receive callbacks

Response (200)

Your callback URL must respond with JSON object `{'status': 'success' }` and HTTP code 200 to Xprizo.

Test Approval Webhook

You can test your callback URL's response by using this endpoint

Request

```
curl --location --request GET '{baseurl}/Preference/TestApprovalWebhook' \  
--header 'Accept: text/plain; x-api-version=1.0' \  
--header 'x-api-version: {apiversion}' \  
--header 'x-api-key: {apikey}'
```

Response (200)

Your callback URL must respond with JSON object `{'status': 'success' }` and HTTP code 200 to Xprizo.

Approval Webhook Responses

Created Transaction & Approved Transaction Response on Webhook (Sample)

```
# Created Transaction  
{  
  "statusType": 1,  
  "status": "New",  
  "description": null,  
  "actionedById": 724,  
  "affectedContactIds": [  
    723,
```

```

    2
  ],
  "transaction": {
    "id": 0,
    "createdById": 724,
    "type": "UCD",
    "date": "2024-02-07T03:41:59.823195+00:00",
    "reference": "652-1706532591283",
    "currencyCode": "USD",
    "amount": 5
  }
}

# Approved Transaction
{
  "statusType": 2,
  "status": "Accepted",
  "description": null,
  "affectedContactIds": [
    723,
    2
  ],
  "transaction": {
    "id": 3169,
    "createdById": 724,
    "type": "UCD",
    "date": "2024-02-07T03:43:50.9186754+00:00",
    "reference": "652-1706532591283",
    "currencyCode": "USD",
    "amount": 5
  }
}

```

Parameter	Description
statusType	This is the status 2 = Accepted
status	Text version of the status "Accepted"

affectedContactIds	A list of the contact IDs that were involved in this transaction
Transaction	Transaction that was created
id	The unique Id of the transaction provided by Xprizo
createdById	Id of the person who creates the transaction
type	The type of transaction (UCD = Card Deposit)
date	The date of the transaction
reference	Recipient's unique transaction identifier for this transaction
currencyCode	The currency of this transaction
amount	The amount of this transaction

Rejected Transaction Response on Webhook (Sample)

```
{
  "statusType": 3,
  "status": "Rejected",
  "description": "Reason for rejection",
  "actionedById": 1,
  "affectedContactIds": []
  "transaction": {
    "id": 0,
    "createdById": 2,
    "type": "UCD",
    "date": "2021-04-20T20:34:00.7606173+02:00",
    "reference": 234234234,
    "currencyCode": "USD",
    "amount": 100.00
  }
}
```

The following is a list of approval status codes that could be returned

- 0 = None - Used for testing

- 1 = New - A new pending transaction has been created
- 2 = Accepted- The transaction was approved and processed
- 3 = Rejected - The transaction was rejected by the acquirer
- 4 = Cancelled - The transaction was canceled by the creator

Update Payment Webhook

Payment Webhook is triggered when a payment transaction with an action set to 2 (callback) is created.

Callbacks are only sent to the profile that created the transaction.

Callbacks will keep sending requests to the platform for 24 hours or until it gets a response.

If a 200 (success) response is received then the transaction will be processed.

If any other response (except a timeout) is received then the transaction will be voided.

Request

```
curl --location --request PUT '{baseurl}/Preference/SetPaymentWebhook?url={url}' \
--header 'Accept: text/plain; x-api-version=1.0' \
--header 'x-api-version: {apiversion}' \
--header 'x-api-key: {apikey}'
```

Parameter	Description
url	Url to receive callbacks

Response (200)

Payment Webhook Responses

This will triggered when the send payment is initiated in Callback mode, Based on the response of this webhook by the Recipient, Xprizo will approve or void the transaction.

```
{
  "id": 3291,
  "currencyCode": "INR",
  "amount": 10,
  "reference": "652-1706532591321",
  "status": "Success"
}
```

Parameter	Description
status	Status of transaction
id	The unique ID of the transaction provided by Xprizo
reference	Recipient's unique transaction identifier for this transaction
currencyCode	The currency of this transaction
amount	The amount of this transaction

Recipient will use this response and check for transactions having the same reference in their system and if everything is fine then they will respond with HTTP status code 200 else they will provide another HTTP status code (excluding timeout).

If the Status code is 200 then this transaction will be marked as Completed in the Xprizo system else it will be VOID (excluding timeout).

Wallet Endpoints (api/Wallet)

General Definitions

Parameter	Description
baseurl	Sandbox - https://test.xprizo.com/api Live- https://wallet.xprizo.com/api
apiversion	1.0
apikey	API key For authentication

Get Wallet Information

Request

```
curl --location --request GET
' {baseurl}/Wallet/Info?contact={username/emailid/mobilenumber}&currencyCode={currencyCode}' \
--header 'Accept: text/plain; x-api-version=1.0' \
--header 'x-api-version: {apiversion}' \
--header 'x-api-key: {apikey}'
```

Parameter	Description
contact	enter the user's mobile number, username, or email for searching
currencyCode	the currency code of the wallet to search for - leave blank to fetch the user's default wallet.

Response (200)

```
{
  "id": <accountId>,
  "contactId": <contactId>,
```

```
"contact": <owners Name>,  
"currencyCode": <currency code>,  
"symbol": <currency symbol>,  
"name": <wallet name>,  
"profilePicture": <owners picture>,  
"isDefault": <is the owners default wallet>,  
"isMerchant": <is the owners have merchant wallet>
```

```
}
```