



## Testing

When integrating a payment gateway, you'll need to test your implementation before going live. We've got test credentials for you to test a variety of use cases and allow you to simulate both successful and failed scenarios.



### Using test credentials and mock data

These credentials and data work exclusively in **Test Mode**. If you require mock data for testing a specific flow or feature, please contact [support](#).

## Testing Payments on Checkout

To properly render checkout on your local environment, you'd need to route your app to `localhost:<preferred-port>` e.g. `localhost:3000`. Pointing your app to an ip address e.g. `127.0.0.1:3000` would result in errors.

Credentials are listed in the **card**, and **mobile money** sections to help you test your checkout integration.

## OTPs

Any OTP passed in test transactions will pass validation. However, you can use these special OTPs to mock specific error scenarios:

- WRONG OTP: 5548
- INSUFFICIENT FUNDS: 6648



These special OTPs will only work (simulate failed payments) when the OTP validation is happening directly in our payment modal. If you are redirected to our OTP validation page, you'll need to use one of the test cards designated for **failed payments**.

## Bank accounts

Bank account details to use to make a mock payment.

Bank: **Access Bank (044)**

Account number: **0690000031**

OTP: **12345**

Bank: **Access Bank (044)**

Account number: **0690000032**

OTP: **12345**

Bank: **Access Bank (044)**

Account number: **0690000033**

OTP: **12345**

Bank: **Access Bank (044)**

Account number: **0690000034**

OTP: **12345**



### Testing tip

If you need more Access Bank test account numbers, you can keep incrementing the last digit of the test account numbers above to get new test account numbers, right up to

0690000041 .

## Mobile money

## Successful payments

To mock a successful mobile money payment, you can use any mobile number.

**Failed payments**

Mock failed transactions for your integration tests using any of the following numbers. Update the country code of each number to match the code of your customer's number.

S/N	Mobile number	Error message
1.	233121212121	Mocked a Failed Transaction
2.	233010101011	Mocked a Failed Transaction

**Cards**

Card details to use to make a mock payment.

**Successful payments**

Use these cards if you want the payment to succeed.

Type	Card number	CVV	PIN	Expiry	C
MasterCard PIN authentication	5531886652142950	564	3310	09/32	1
Mastercard PIN	5399838383838381	470	3310	10/31	1

Type	Card number	CVV	PIN	Expiry	C
authentication 2					
MasterCard 3DS authentication	5438898014560229	564	3310	10/31	1
Visa Card 3DS authentication	4187427415564246	828	3310	09/32	1
Visa Card 3DS authentication 2	4242424242424242	812	3310	09/31	1
Visa Card 3DS authentication 3	4751763236699647	470	-	09/35	-
Verve Card Noauth	5061460410120223210	780	3310	10/31	1
Verve Card PIN authentication 2	5061460166976054667	564	3310	10/29	-
Address Verification (AVS) Card	4556052704172643	899	3310	09/32	1
Pre- authorization	5377283645077450	789	3310	09/31	-

Type	Card number	CVV	PIN	Expiry	C
Test Card					

Failed payments

Use these cards if you want the payment to fail.

Type	Card number	CVV	PIN	Expiry	OTP
Card Declined (Address Verification)	5143010522339965	276	3310	08/32	12345
Card Fraudulent	5590131743294314	887	3310	11/32	12345
Card Insufficient Funds	5258585922666506	883	3310	09/31	12345
Do Not Honour	5143010522339965	276	3310	08/31	-
Insufficient Funds	5258585922666506	883	3310	09/31	12345
Invalid Transaction	5551658157653822	276	-	08/31	-

Type	Card number	CVV	PIN	Expiry	OTP
Restricted Card, Retain Card	5551651630381384	276	-	08/31	-
Function Not Permitted to Cardholder	5258582054729020	887	-	11/30	-
Function Not Permitted to Terminal	5258588264565682	887	-	11/30	-
Transaction Error	5258589130149016	887	-	11/30	-
Incorrect PIN	5399834697894723	883	3310	09/31	12345
Verve Card - Card enrolment	5531882884804517	564	3310	10/32	-

## Testing transfers

When testing a transfer, you can use any of our provided [test accounts](#). By default, such transfers will always remain in a `PENDING` state. You can force transfers to behave differently by using a special kind of transaction reference when creating the transfer.

- **To mock a successful transfer:** Make sure your reference ends with `_PMCK` (for example, `"dfs23fhr7ntg0293039_PMCK"` ).
- **To mock a failed transfer:** Make sure your reference ends with `_PMCK_ST_F` (for example, `"dfs23fhr7ntg0293039_PMCK_ST_F"` ).

By default, the status of the mocked transfer will only be updated after 10 minutes.

- **To change the time delay:** append `DU_{minutes}` to your transaction reference.

For example:

- a transfer with a reference of `"dfs23fhr7ntg0293039_PMCK"` will succeed after 10 minutes
- a transfer with a reference of `"dfs23fhr7ntg0293039_PMCKDU_1"` will succeed after 1 minute
- a transfer with a reference of `"dfs23fhr7ntg0293039_PMCK_ST_F"` will fail after 10 minutes
- a transfer with a reference of `"dfs23fhr7ntg0293039_PMCK_ST_FDU_1"` will fail after 1 minute

## Bill Payments

Here are some credentials to help you test some billers

Billers	Credential Type	Credentials
DSTV	Smart card number	0025401100
EKEDC	Meter number	1101160434535



### DSTV tests

The Biller type for making mock DSTV payments is **DSTV Payment**

## BVN Credentials

Use these mock data to test your BVN consent integration

BVN: **22222222280**

BVN: **22123456789**

First Name: **Nibby**

First Name: **Nibby**

Last Name: **Certifier**

Last Name: **Certifier**

OTP: **111111**

Did you find this page helpful?

Yes I did 😊

Not exactly 😞



### Have any questions?

Get in touch if you have any question or need help with integrating the Flutterwave API.



### Video tutorials

Check out our YouTube channel for tutorials on how to integrate the Flutterwave API.