CRONFLY - Documentation for RESTful API (version 2.0)

URL/Endpoint: https://cronfly.com/cronapis (POST)

#	Transaction	Request type
1	Create Temporal Account	POST
2	Create Dedicated Account	POST
3	Initiate Payout Transfer	POST
4	Query Transaction by Reference	POST
5	Query Transaction by Session ID	POST
6	Fetch Transaction Range	POST
7	Update Permanent Account	POST
8	Webhook Listening	-
9	Standard Online Payment*	POST

Headers	Value
Api-Key	API Key as provided
Merchant-Signature	sha512(APIKEY:MerchantID)

PAYMENT ACCOUNT OPTIONS OFFERED BY CRONFLY

1.   The Merchant's Permanent Account

- This account is issued to the Merchant only
- Users have to register their account numbers as payment accounts on their dashboard before transferring funds to the Merchant's Permanent Account
- Each user can also register as many payment accounts as he/she may need
- Users are to transfer funds only from their registered payment account numbers to The Merchant's Permanent Account using their bank's mobile app or USSD code.
- You may not be able to give instant value to customers for payments from unregistered account numbers, POS, ATM, Bank deposits or transfers through *third party apps* like OPAY, CARBON, NOMBA etc.

2.   The User's Temporal Account

- This account is valid for 24hours only (expires after 24 hours)
- This account is a one-time payment account as it **accepts only one payment**
- Users can generate as many temporal accounts as they may need per day but are to only pay once into each account generated
- Payments can be made into the temporal accounts from all banks(including Microfinance banks), using POS, ATM, Bank transfers or deposits and through third party apps like OPAY, CARBON, NOMBA etc.

3.   The User's Dedicated Account

- This account is a reserved account for the User only
- Each user is required to provide his/her BVN before this account is issued to him/her
- Payments can be made into the User's Dedicated Account from all banks (including Microfinance banks), using POS, ATM, Bank transfers or deposits and through third party apps like OPAY, CARBON, NOMBA etc.

NOTE: The Merchant's BVN is tied to each User's Dedicated Account issued;
- You are hereby advised to ensure that your customers do not use the account number for fraudulent activities as it is against CBN's Anti-money laundering (AML) policy.

Transaction	Headers / Parameters

Create Temporal Account	-- H "Api-Key: YOUR_API_KEY"
	-- H "Merchant-Signature: MERCHANT_SIGNATURE"
	-- d '{ "requestType": "CreateTemporalAccount", "accountName": "Shitu Omolara", "customerMail": "shitulara_f_s@ymail.com", "requestReference": "crfly_fr56wyr6ehe"}'
	
	Success Response
	
	{
	
	"status":"Successful",
	
	"statusCode":200,
	
	"message":"Temporal account creation was successful.",
	
	"accountNumber":"0123456789",
	
	"accountName":"Cronfly- Shitu Omolara ",
	
	"accountExpiresOn":"2020-06-11 19:15:27",
	
	"customerMail":"shitulara_f_s@ymail.com",
	
	"requestRef":"crfly_fr56wyr6ehe"
	
	"creationReference":"CronFly20r56wyr6ehe10d"
	
	}
	
	• **Credit Customer's wallet by mapping [Customer's Mail <u>in Webhook</u>] to [Stored E-mail Address <u>in Database</u>]**
	
	• **NOTE: Temporal Accounts expires after 24 hours and can/should only receive payment ONCE.**

Create Dedicated Account	-- H "Api-Key: YOUR_API_KEY"
	-- H "Merchant-Signature: MERCHANT_SIGNATURE"

-- d '{ "requestType": "CreateDedicatedAccount", "accountName": "Shitu Omolara", "customerMail": "shitulara_f_s@ymail.com", "customerBVN": "22201234567", "requestReference": "crfly_fr56wyr6ehe"}'

Success Response

{

"status":"Successful",

"statusCode":200,

"message":"Dedicated account creation was successful.",

"accountNumber":"0123456789",

"accountName":"Cronfly- Shitu Omolara ",

"customerMail":"shitulara_f_s@ymail.com",

"bankName":"Providus Bank",

"bankCode":"01",

"requestRef":"crfly_fr56wyr6ehe"

"creationReference":"CronFly20r56wyr6ehe10d"

}

- **Credit Customer's wallet by mapping [Customer's Mail in Webhook] to [Stored E-mail Address in Database]**

- **NOTE: Temporal Accounts expires after 24 hours and can/should only receive payment ONCE.**

Initiate Payout Transfer	-- H "Api-Key: YOUR_API_KEY"
	-- H "Merchant-Signature: MERCHANT_SIGNATURE"

-- d '{ "requestType": "PayoutTransfer", "requestReference": "crfly_fr56wyr6ehe10d"}'

Success Response

{

"status":"Successful",

"statusCode":200,

"message":"Payout request was successful.",

"requestRef":"crfly_fr56wyr6ehe10d"

}

Query Transaction by Reference	-- H "Api-Key: YOUR_API_KEY"
	
	-- H "Merchant-Signature: MERCHANT_SIGNATURE"
	
	-- d '{ "requestType": "QueryByReference", "paymentReference": " CRFLY|01|THU20052021000001"}'
	
	Success Response
	{
	"status":"Successful",
	"statusCode":200,
	"paymentRef":"CRFLY|01|THU20052021000001",
	"sessionId":"202210521000074900002",
	"accountNumber":"0123456789",
	"amountPaid":"100",
	"settledAmount":"97",
	"transactionFee":"3",
	"currency":"NGN",
	"depositorAccountNumber":"9876543210",
	"depositorName":"SHITTU OMOLARA A.",
	"bankName":"GT BANK",
	"transactionDate":"2020-05-20",
	"paymentNarration":"ONLINE TRF FROM CRONFLY TO JJJ AND CO.",
	"clientKey":"fad0988dcd4e685f27d85d0e787f47d2db4873115fa82abbb3793454b21a0c02b89d13f07b4b6c4d339c07a02b5d4c44d5587f777bfeabcef5856133970500af",
	"notificationsSent":"YES",
	"webhookFeedback":
	"{
	"responseCode":100,
	"sessionId":null,
	"responseMessage":"transaction received"
	}"
	}

Query Transaction by Session ID	-- H "Api-Key: YOUR_API_KEY" -- H "Merchant-Signature: MERCHANT_SIGNATURE" -- d '{ "requestType": "QueryBySessionID", "sessionId": "202210521000074900002"}'

Success Response

```
{
"status":"Successful",
"statusCode":200,
"paymentRef":"CRFLY|01|THU20052021000001",
"sessionId":"202210521000074900002",
"accountNumber":"0123456789",
"amountPaid":"100",
"settledAmount":"97",
"transactionFee":"3",
"currency":"NGN",
"depositorAccountNumber":"9876543210",
"depositorName":"SHITTU OMOLARA A.",
"bankName":"GT BANK",
"transactionDate":"2020-05-20",
"paymentNarration":"ONLINE TRF FROM CRONFLY TO JJJ AND CO.",
"clientKey":"fad0988dcd4e685f27d85d0e787f47d2db4873115fa82abbb3793454b21a0c0
2b89d13f07b4b6c4d339c07a02b5d4c44d5587f777bfeabcef5856133970500af",
"notificationsSent":"YES",
"webhookFeedback":
"{
"responseCode":100,
"sessionId":null,
"responseMessage":"transaction received"
}"
}
```

Fetch Transaction Range	-- H "Api-Key: YOUR_API_KEY"
	-- H "Merchant-Signature: MERCHANT_SIGNATURE"
	-- d '{ "requestType": "FetchTransactions", "bankCode": "00", "startDate":"TODAY", "endDate":"", "requestReference": "crfly_fr56wyr6ehe10d"}'

Success Response

```
{"status":"Successful",
"statusCode":200,
"message":"Transaction query was successful.",
"bankCode":"00",
"requestRef":"gioljjh7qwYYU55166728",
"transactions":
[{
"paymentRef":"CRFLY|01|THU20052021000001",
"sessionId":"202210521000074900002",
"accountNumber":"0123456789",
"amountPaid":"100",
"settledAmount":"97",
"transactionFee":"3",
"currency":"NGN",
"depositorAccountNumber":"9876543210",
"depositorName":"SHITTU OMOLARA A.",
"bankName":"GT BANK",
"transactionDate":"2020-05-20",
"paymentNarration":"ONLINE TRF FROM CRONFLY TO JJJ AND CO.",
"clientKey":"fad0988dcd4e685f27d85d0e787f47d2db4873115fa82abbb3793454b21a0c0
2b89d13f07b4b6c4d339c07a02b5d4c44d5587f777bfeabcef5856133970500af",
"notificationsSent":"YES",
"webhookFeedback":"{"responseCode":100, "sessionId":null,
"responseMessage":"transaction received"}"
}]
}
```

API Constants (Know the constants before consuming the API)

#	API Constants	Constant Values	Descriptions
	requestType	**CreateTemporalAccount**	To create a Temporal Account (expires in 24 hours)
		QueryBySessionID	To Query transaction using Session ID
		QueryByReference	To Query transaction using Payment Reference
		PayoutTransfer	To initiate a Payout Transfer
		FetchTransactions	To fetch transactions based on Dates
1	*bankCode*	**00**	Providus Bank
		04	Cronfly Wallet
		05	Cronfly Voucher
		06	Cronfly Virtual card
2	*startDate*	**TODAY**	Current Date API is Called

API Parameters

Parameter/Field name	Type	Description
requestType *	String	*(API Constant ↑)* This describes the kind of transaction you want to initiate
requestReference *	String	Your unique reference id used to identify each API call (50 chars Max.)
sessionId *	String	Payment Session ID as provided by the Bank to Query a transaction
paymentReference *	String	Unique Payment Reference to Query a transaction by Reference
bankCode *	String	*(API Constant ↑)* This describes the bank to which you are making an API call
startDate *	String/Timestamp	*(API Constant ↑)* Can be passed as "TODAY" or as UNIX Timestamp while fetching transactions
endDate *	Timestamp	UNIX Timestamp to identify the end date for fetched transactions
accountName *	String	This is name of Temporal Account to be created
customerMail *	String	E-mail address of customer for which Temporal Account is to be created
customerBVN *	String	Biometric Verification Nnumber of customer for which Dedicated Account is to be created

API Response Codes – You get one of these responses when you initiate a transaction via the API

#	Status codes	Status	Message
1	200	Successful	Request was successful
2	000	Failed	Request not provisioned
3	001	Failed	IP address not whitelisted
5	002	Failed	Insufficient privilege
6	201	Failed	Incomplete credentials
7	202	Failed	Invalid credentials.
8	203	Failed	Invalid request type.
9	204	Failed	Insufficient wallet balance.
10	205	Failed	Invalid or duplicate reference
11	206	Failed	Incomplete parameters
12	207	Failed	Invalid bank code
13	209	Failed	Authentication failed
14	210	Failed	Technical error! Request failed
15	211	Failed	Unknown Error: Unable to process request
16	212	Failed	Invalid Session ID

Payment Webhook Listening

Process flow	**Steps [1 – 4] are mandatory for Instant Funding with a PERMANENT/LINKED Account**
	1. Store/Add Customer's Payment Account (Account Number) **to Database**
	2. Display Merchant's Account number to Customer for payment (After Payment Account is Added)
	3. **Credit Customer's wallet by mapping [Depositor's Account Number <u>in Webhook</u>] to [Stored Account Number <u>in Database</u>]**
	4. *You may hide Merchant's Account from members at first and make it visible after [1] Payment Account is Added*
	NOTE: Customer can pay into a **TEMPORAL ACCOUNT** without having to add/register a payment account
Test Parameters	**Api-Key:** apk_test_a97ffaca5a88d3825abdb40b23e3642c169ff939
	Webhook-Key: wbh_test_5652101aaa9bb3aa754c2c6f9edc7d1ea51c55fe
	MerchantID: 011
	ClientKey: sha512(APIKEY:MERCHANTID:SESSIONID)
	[Replace TESK KEYS above with LIVE KEYS for Production/Live use]
	NOTE: Webhook payload is accompanied with only one header: **WEBHOOK-KEY** while the **ClientKey** is sent as part of the payload body
	[Please confirm the headers match one or both of the KEYS above before giving value]

Webhook Payload	<pre>{
	"paymentRef":"CRFLY|01|THU20052021000001",
	"sessionId":"202210521000074900002",
	"accountCredited":"0123456789",
	"amountPaid":"100",
	"settledAmount":"97",
	"transactionFee":"3",
	"currency":"NGN",
	"depositorAccountNumber":"9876543210",
	"depositorName":"SHITTU OMOLARA A.",
	"bankName":"GT BANK",
	"transactionDate":"2020-05-21 02:29:01 PM",
	"paymentNarration":"ONLINE TRF FROM CRONFLY TO JJJ AND CO.",
	"clientKey":"f4049276794cd04b814acb992bf92c391a0ab38696ac948d274ea9a1a82704efdd5baefe",
	"customerMail":"johnfamous@gmail.com",
	"transactionType":"CREDIT",
	"transactionPurpose":"WALLET"
	}</pre>

Responses
When our payment payload hits your server (via Webhook URL), we expect you to send us <u>one</u> of the following **JSON** responses, depending on the status of the transaction at your end.

For success
```
{
"responseCode": 100,
"sessionId": "202210521000074900002",
"responseMessage": "transaction received"
}
```

For rejected transaction
```
{
"responseCode": 200,
"sessionId": "202210521000074900002",
"responseMessage": "transaction rejected"
}
```

For duplicate transaction
```
{
"responseCode": 300,
"sessionId": "202210521000074900002",
"responseMessage": "duplicate transaction"
}
```

For further guide, support or inquiry on consumption of this API, please contact us via support@cronfly.com

<u>Last Reviewed</u>: Wednesday September 14, 2022