Orange API Documentation

External Cash In-Out APIs

Abstract

These APIs support secure wallet operations through agents and external channels.

Orange Money Technical Team

orangemoneytechnical.ojo@orange.com

Version 1.0.0

05 Nov 2015

Orange API Documentation

This document contains a collection of APIs that serve External Cash In-Out functionalities:

- **Get Wallet Details: Retrieves detailed wallet information.**
- **Proccess Transaction External Cash In: Credits (adds funds to) a customer's wallet when cash is deposited via an external/agent channel.**
- **Proccess Transaction External Cash Out: Debit a customer wallet and pay out cash via an external/agent channel.**

1. Introduction

These APIs provide the core external channel functionalities, enabling agents and partners to securely manage customer wallets. They cover three main operations: funding a wallet (Cash-In), withdrawing from a wallet (Cash-Out), and retrieving wallet details.

Base URL (Production): https://orangemoney.orange.jo:1594/

Base URL (Sandbox): https://om-dev.orange.jo:1445/

2. Test requirement

To ensure a smooth integration and testing processes. The following points need to be considered for testing and production:

1. **Two phone numbers are required for testing purposes:**
   - **One associated with the Personal Wallet.**
   - **One associated with the Business Wallet.**
2. **Credentials will be sent by Orange Money through text message on the number related to business wallet.**
3. **Credentials for testing (staging) are different from the ones for live (production).**
4. **The agent needs to provide a static IP address that needs to be whitelisted by Orange Money during testing (staging) and another one for live (production).**
5. **The agent can review all transactions from the Agent Portal.**

3. Authentication

All API endpoints require JWT token–based authentication to be included in the Authorization header:

- **Authorization: Bearer <AccessToken>.**

📌 ***Notes:***

- The external client must first authenticate using the Authorization API, which returns a JWT token.

Authorization

Path	api/ExternalAPI/Authorization
HTTP Method	POST
JSON format	Content-Type: application/json

Request

Parameter	Type	Required	Description	Encryption
UserName	string	Yes	The UserName	Yes
Password	string	Yes	The Password	Yes

Response

Parameter	Type	Description
AccessToken	String	Access Token
token_type	String	Toke type always Bearer
expires_in	int	ExpiresIn
errorCode	String	Error code
errorDescription	String	Error message text
isSuccess	bool	Indicates if the transaction was successful
IsOTPRequired	bool	Specifies if an OTP is required for the transaction
errors	List [{ "description": "string",	List of Errors

	"descriptionAr ": "string", }]	

4. API Structure & Design

GetWalletDetailsV2

Path	api/ExternalAPI/GetWalletDetailsV2
HTTP Method	POST
JSON format	Content-Type: application/json

ProcessTransactionExternalCashInV2

Path	api/ExternalAPI/ProcessTransactionExternalCashInV2
HTTP Method	POST
JSON format	Content-Type: application/json

ProcessTransactionExternalCashOutV2

Path	api/ExternalAPI/ProcessTransactionExternalCashOutV2
HTTP Method	POST
JSON format	Content-Type: application/json

5. Error Handling

The API's use standard HTTP status codes:

Code	Meaning

200	OK
500	Internal Server Error

The API's also return the following Application-Level errors codes:

📌 ***Note: these error codes are returned in the response with 200 OK HTTP status code.***

Code	Description	Description Arabic
17	Invalid One Time Password	كلمة المرور لمرة واحدة غير صحيحة
18	One Time Password is expired	انتهت صلاحية كلمة المرور لمرة واحدة
35	Sender Wallet Not Exist	محفظة الارسال غير موجودة
36	Receiver Wallet Does Not Exist	محفظة الإستقبال غير موجودة
38	Sender Wallet Is Freezed	محفظة المرسل معطلة
39	Receiver Wallet Is Freezed	محفظة الإستقبال معطلة
41	Insufficient Balance We cannot process this transaction as the wallet balance is not enough	الرصيد غير كاف لا يمكن إتمام هذه العملية. رصيد المحفظة غير كاف
44	Cannot continue this process, the Sender exceeded the allowed number of attempts, please try again later	لا يمكن إكمال هذه العملية، لقد تجاوز المرسل عدد المحاولات المسموح بها، يرجى المحاولة لاحقاً
45	Can not continue this process, the receiver exceeded the allowed number of attempts, please try again later	لا يمكن إكمال هذه العملية، لقد تجاوز المستقبل عدد المحاولات المسموح بها، يرجى المحاولة لاحقاً
46	Transaction amount exceeds the allowed limit	قيمة هذه الحركة أعلى من الحد المسموح
47	Sender wallet max limit has been exceeded	تم تجاوز الحد الأقصى لمحفظة الارسال
48	Receiver wallet max limit has been exceeded	تم تجاوز الحد الأقصى لمحفظة الاستقبال
53	Transaction is not allowed	المعاملة غير مسموح بها

106	**Please choose the appropriate transaction type**	خطأ غير متوقع. نوع المعاملة مطلوب
120	**Can not continue this process, Sender Wallet Is Dormant**	لا يمكن متابعة هذه العملية، محفظة المرسل غير نشطة
120	**Can not continue this process, Receiver Wallet Is Dormant**	لا يمكن متابعة هذه العملية، محفظة المستقبل غير نشطة
170	**OTP is Required**	كلمة المرور لمرة واحدة مطلوبة
230	**Invalid Signature**	**Invalid Signature**
236	**The amount entered is invalid**	المبلغ المدخل غير صحيح
288	**As per our internal policy we cannot proceed the transaction**	بناءً للسياسة الداخلية لا يمكن تنفيذ الحركة
303	**Can not continue this process, Sender Wallet exceeded number of allowed debit transaction per Day**	لا يمكن متابعة هذه العملية، حيث تجاوزت محفظة المرسل عدد المعاملات المسموح بها للخصم في اليوم
304	**Can not continue this process, Sender Wallet exceeded allowed amount of debit transaction per Day**	لا يمكن متابعة هذه العملية، حيث تجاوزت محفظة المرسل المبلغ المسموح به لمعاملات السحب في اليوم
305	**Can not continue this process, Sender Wallet exceeded allowed amount of debit transaction per month**	لا يمكن متابعة هذه العملية، حيث تجاوزت محفظة المرسل المبلغ المسموح به لمعاملات السحب في الشهر
311	**The minimum allowed amount is 0.1 JD**	الحد الأدنى للمبلغ المسموح به هو **0.1** دينار

6. Endpoints

6.1 GetWalletDetailsV2

To retrieve wallet details by MSISDN after agent authentication and OTP verification. Designed for external/partner use.

The transaction process is carried out in two steps:

1. Validation Step: Send the request with IsConfirmed = false. This is used to validate the wallet details.
2. Execution Step: Send the same request again with IsConfirmed = true and include the OTP. to returns the wallet details.

Method	**POST**
URL	**api/ExternalAPI /GetWalletDetailsV2**
Header	**Key: Signature** **Value: “string”**
Request	**{** **"MSISDN": "string (required, encrypted)",** **"IsConfirmed": "boolean (required, not encrypted)",** **"OTP": "string (optional/empty, encrypted if provided)",**
Response	**{** **"WalletId": 0,** **"Identity": "string",** **"CustomerName": "string",** **"StatusId": 0,** **"StatusText": "string",** **"Balance": 0.0,** **"isSuccess": true,** **"errorCode": "string",** **"errorDescription": "string",** **"errorArDescription": "string",** **"errors": [** **{** **"description": "string",** **"descriptionAr": "string"** **}** **]** **}**

6.2 ProcessTransactionExternalCashInV2

Credit a customer wallet with cash collected by an external/agent channel.

The transaction process is carried out in two steps:

1. **Validation Step: Send the request with IsConfirmed = false. This is used to validate the transaction details before execution. This also includes sending OTP to the customer.**
2. **Execution Step: Send the same request again with IsConfirmed = true and including the OTP to finalize and perform the transaction.**

Method	**POST**
URL	**api/ExternalAPI/ProcessTransactionExternalCashInV2**
Header	**Key: Signature** **Value: "string"**
Request	**{** **"MSISDN": "string (required, encrypted)",** **"TransactionNote": "string (required, encrypted)",** **"TellerUserName": "string (required, encrypted)",** **"TellerName": "string (required, encrypted)",** **"Amount": "string (required, encrypted)",** **"IsConfirmed": "boolean (required, not encrypted)",** **"OTP": "string (required, encrypted)"** **}**
Response	**{** **"WalletUserName": "string",** **"MSISDN": "string",** **"TransactionAmount": 0,** **"TransactionFees": 0,** **"TransactionTotalAmount": 0,** **"WalletBalanceBefore": 0,** **"WalletBalanceAfter": 0,**

```
"TransactionId": 0,
"OJPaymentId": 0,
"transactionReference": "string",
"merchantName": "string",
"TransactionTypeId": 0,
"TransactionTypeArName": "string",
"TransactionTypeEnName": "string",
"ReceiverName": "string",
"TransactionTip": 0,
"ServicerNameEn": "string",
"ServicerNameAr": "string",
"SenderPhonerNumber": "string",
"ReceiverPhonerNumber": "string",
"FeesWalletPhonerNumber": "string",
"isSuccess": true,
"IsOTPRequired": true,
"errors": [
  {
    "description": "string",
    "descriptionAr": "string"
  }
 ],
}
```


Note: **Must call API Get Wallet Details then this API with the OTP that arrived.**

6.3 ProcessTransactionExternalCashOutV2

Debit a customer wallet and pay out cash via an external/agent channel

The transaction process is carried out in two steps:

1. **Validation Step: Send the request with IsConfirmed = false. This is used to validate the transaction details before execution. This also includes**

sending OTP to the customer.

2. Execution Step: Send the same request again with IsConfirmed = true and including the OTP to finalize and perform the transaction.

Method	POST
URL	api/ ExternalAPI/ProcessTransactionExternalCashOutV2
Header	Key: Signature Value: “string”

Request	{
	"MSISDN": "string (required, encrypted)",
	"TransactionNote": "string (required, encrypted)",
	"TellerUserName": "string (required, encrypted)",
	"TellerName": "string (required, encrypted)",
	"Amount": "string (required, encrypted)",
	"IsConfirmed": true,
	"OTP": "string (required, not encrypted)"
	}

Response	{ "WalletUserName": "string", "MSISDN": "string", "TransactionAmount": 0, "TransactionFees": 0, "TransactionTotalAmount": 0, "WalletBalanceBefore": 0, "WalletBalanceAfter": 0, "TransactionId": 0, "OJPaymentId": 0, "transactionReference": "string", "merchantName": "string", "TransactionTypeId": 0, "TransactionTypeArName": "string", "TransactionTypeEnName": "string", "ReceiverName": "string",

```
"TransactionTip": 0,
"ServicerNameEn": "string",
"ServicerNameAr": "string",
"SenderPhonerNumber": "string",
"ReceiverPhonerNumber": "string",
"FeesWalletPhonerNumber": "string",
"isSuccess": true,
"IsOTPRequired": true,
"errors": [
  {
    "description": "string",
    "descriptionAr": "string"
  }
 ],
}
```


Note: **Must call API Get Wallet Details then this API with the OTP that arrived.**

3. Data Models

a. **GetWalletDetailsV2**

7.1.1 Request

Parameter	Type	Required	Description	Encryption
MSISDN	string	Yes	The phone number of the user	Yes
isConfirmed	bool	Yes	false just for validation/ true to perform the transaction	No
OTP	string	No	The One-Time Password	Yes

7.1.2 Response

Field	Type	Description
WalletId	integer	Wallet Id
Identity	string	Customer ID
CustomerName	string	Customer Name
StatusId	integer	Wallet Status Id
StatusText	string	Wallet Status Text
Balance	decimal	Balance
isSuccess	bool	Indicates if the transaction was successful
errorCode	string	Error code
errorDescription	string	English Error
errorArDescription	string	Arabic Error
errors	List [{ "description" : "string", "description Ar": "string", }]	List of Errors

b. ProcessTransactionExternalCashInV2

7.2.1 Request

Parameter	Type	Required	Description	Encryption
MSISDN	string	Yes	The phone number of the user	Yes
TransactionNote	string	Yes	The transaction note	Yes
TellerUserName	string	Yes	The Teller User Name	Yes
TellerName	string	Yes	The User Name	Yes
Amount	string	Yes	The amount of the transaction	Yes
isConfirmed	bool	Yes	false just for validation/	No

			true to perform the transaction	
OTP	string	Yes	The One-Time Password	Yes

7.2.2 Response

Parameter	Type	Description
WalletUserName	String	The username associated with the wallet.
MSISDN	String	The MSISDN is phone number
TransactionAmount	Decimal	The amount of the transaction
TransactionFees	Decimal	The fees associated with the transaction.
TransactionTotalAmount	Decimal	The total amount of the transaction, including fees.
WalletBalanceBefore	Decimal	The balance in the wallet before the transaction occurred.
WalletBalanceAfter	Decimal	The balance in the wallet after the transaction occurred.
TransactionId	Integer	The unique identifier for the transaction.
OJPaymentId	Integer	The payment ID for the transaction
transactionReference	String	The reference ID for the transaction
merchantName	String	The name of the merchant involved in the transaction.
TransactionTypeId	Integer	The ID representing the type of transaction
TransactionTypeArName	String	The name of the transaction type in Arabic.
TransactionTypeEnName	String	The name of the transaction type in English.
ReceiverName	String	The name of the receiver for the transaction.
TransactionTip	Decimal	A tip or additional amount given during the transaction
ServicerNameEn	String	The name of the service provider in English.
ServicerNameAr	String	The name of the service provider in Arabic.

SenderPhoneNumber	String	The phone number of the sender
ReceiverPhoneNumber	String	The phone number of the receiver
FeesWalletPhoneNumber	String	The phone number associated with the fees wallet
isSuccess	bool	Indicates if the transaction was successful
IsOTPRequired	bool	Specifies if an OTP is required for the transaction
errors	List [{	errors

c. ProcessTransactionExternalCashOutV2

7.3.1 Request

Parameter	Type	Required	Description	Encryption
MSISDN	string	Yes	The phone number of the user	Yes
TransactionNote	string	Yes	The transaction note	Yes
TellerUserName	string	Yes	The Teller User Name	Yes
TellerName	string	Yes	The User Name	Yes
Amount	string	Yes	The amount of the transaction	Yes
isConfirmed	bool	Yes	false just for validation/ true to perform the transaction	No
OTP	string	Yes	The One-Time Password	Yes

7.3.2 Response

Parameter	Type	Description

WalletUserName	String	The username associated with the wallet.
MSISDN	String	The MSISDN is phone number
TransactionAmount	Decimal	The amount of the transaction
TransactionFees	Decimal	The fees associated with the transaction.
TransactionTotalAmount	Decimal	The total amount of the transaction, including fees.
WalletBalanceBefore	Decimal	The balance in the wallet before the transaction occurred.
WalletBalanceAfter	Decimal	The balance in the wallet after the transaction occurred.
TransactionId	Integer	The unique identifier for the transaction.
OJPaymentId	Integer	The payment ID for the transaction
transactionReference	String	The reference ID for the transaction
merchantName	String	The name of the merchant involved in the transaction.
TransactionTypeId	Integer	The ID representing the type of transaction
TransactionTypeArName	String	The name of the transaction type in Arabic.
TransactionTypeEnName	String	The name of the transaction type in English.
ReceiverName	String	The name of the receiver for the transaction.
TransactionTip	Decimal	A tip or additional amount given during the transaction
ServicerNameEn	String	The name of the service provider in English.
ServicerNameAr	String	The name of the service provider in Arabic.
SenderPhoneNumber	String	The phone number of the sender
ReceiverPhoneNumber	String	The phone number of the receiver
FeesWalletPhoneNumber	String	The phone number associated with the fees wallet
isSuccess	bool	Indicates if the transaction was successful
IsOTPRequired	bool	Specifies if an OTP is required for the transaction
errors	List [{	errors

4. Security

To ensure secure communication and data integrity between client systems and the API's, the followings must be followed:

a. Encryption:

The encryption algorithm used is Advanced Encryption Standard (AES) with the flowing Keys:

	Staging	Production
IV	twcny10q3ve3dnbf	r7eAw&u#f40mqYgO
Secret Key	op67yxshux5gyjrasupxqdzwabxyd88s	R4awW6t*fjh4)uZ1xIB$wq)ww^o2sehf

b. Signature (From Header) :

The signature field is used to validate the authenticity of the request. The signature is generated using the SHA-256 hashing algorithm.

📌 *Note: use this link https://emn178.github.io/online-tools/sha256.html*

To generate the signature, use the following formulas per each API:

	Signature Formula
GetWalletDetailsV2	(API KEY + MSISDN + Is Confirmed(True/False) + OTP +API KEY)
ProcessTransaction ExternalCashInV2	(API KEY + Amount +Is Confirmed(True/False) + MSISDN + OTP + TransactionNote + API KEY)
ProcessTransaction ExternalCashOutV2	(API KEY + Amount +Is Confirmed(True/False) + MSISDN + OTP + TransactionNote +API KEY)

API KEY: ABC123

📌 *Note: All the above signature values must be generated before hashing and/or encryption.*

5. Contact Information

For support or questions, please contact:

Email: orangemoneytechnical.ojo@orange.com

Phone: