
mpesawrapper Documentation

Release 1.0.0

Benson Nguru Mburu

Jul 28, 2018

Contents

| | | |
|-----------|--|-----------|
| 1 | Features | 3 |
| 2 | Getting Started | 5 |
| 3 | Installation | 7 |
| 4 | OAUTH | 9 |
| 5 | C2B | 11 |
| 6 | Lipa Na M-Pesa Online Payment API | 13 |
| 7 | B2B | 15 |
| 8 | Transaction Status Request | 17 |
| 9 | Account Balance | 19 |
| 10 | B2C | 21 |
| 11 | Reversal | 23 |
| 12 | Test Credentials | 25 |

Here is a simple guide to help you get started with the daraja API.

Note:

- recommended python version is python3

Install python3 enviroment

- `virtualenv -p python3 envname`
 - `source /path/to/envname/bin/activate`
-

NOTE

This api is still on beta version consult with safaricom LTD before moving to production

CHAPTER 1

Features

The following MPESA features are covered in this package

Note:

- OAUTH - Gives you time bound access token to call allowed APIs
 - C2B -Register URL for Validation/Confirmation as an active listener for MPESA transactions on a shortcode.
 - Lipa Na M-Pesa Online - online payment using STK Push.
 - B2B - Mpesa Transaction from one company to another
 - Transaction Status -Use this API to check the status of transaction
 - Account Balance -Use this API for balance inquiry
 - B2C -Mpesa Transaction from company to client
 - REVERSAL -API to reverse MPESA transaction
-

CHAPTER 2

Getting Started

CHAPTER 3

Installation

```
$ pip install mpesawrapper
```

Import library

```
import mpesawrapper
```


CHAPTER 4

OAUTH

This API should ideally be called before every transaction to obtain an access token for other transactions:

```
consumer_key=consumer key from mpesa
consumer_secret=consumer secret from mpesa
token=mpesawrapper.Generatetoken().generate_token(key=consumer_key,secret=consumer_
↪secret)
```

This token will later be passed in other functions

This module is split in two parts

- Register URL
- Transaction Listener

1. Register URL

```
token = token generated from **OAUTH**
reg_url=mpesawrapper.C2B().register_url(ValidationURL="", ConfirmationURL="",
↳ResponseType="", ShortCode="", token=str(token))
```

Note:

1. ValidationURL for the client URL <https://ip> or domain:port/path
2. ConfirmationURL for the client URL <https://ip> or domain:port/path
3. ResponseType Default response type for timeout. Incase a transaction times out, Completed
4. ShortCode The short code of the organization. Numeric 123456

2. Transaction Listener

```
token = token generated from **OAUTH**
c2b=mpesawrapper.C2B().c2b_transact(ShortCode="", CommandID="CustomerPayBillOnline
↳", Amount=10, Msisdn=254708374149, BillRefNumber="019903023", token=token)
```

Note:

1. CommandID - Unique command for each transaction type. For C2B default String
 - CustomerPayBillOnline
 - CustomerBuyGoodsOnline

2. Amount The amount being transacted Numeric 100
 3. Msisdn Phone number (msisdn) initiating the transaction Numeric MSISDN(12 digits) - 254XXXXXXXXXX
 4. BillRefNumber Bill Reference Number (Optional) Alpha-Numeric Alpha-Numeric less then 20 digits
 5. ShortCode Short Code receiving the amount being transacted Numeric Shortcode (6 digits) - XXXXXX
-

Lipa Na M-Pesa Online Payment API

Note:

1. BusinessShortCode The organization shortcode used to receive the transaction Numeric Shortcode (6 digits)
 2. Password The password for encrypting the request String `base64.encode(ShortcodePasskeyTimestamp)`
 3. Timestamp The timestamp of the transaction Timestamp `yyyymmddhhiiss`
 4. TransactionType The transaction type to be used for the request 'CustomerPayBillOnline' String `CustomerPay-BillOnline`
 5. Amount The amount to be transacted Numeric 1
 6. PartyA The entity sending the funds Numeric MSISDN (12 digits)
 7. PartyB The organization receiving the funds Numeric Shortcode (6 digits)
 8. PhoneNumber The MSISDN sending the funds Numeric MSISDN (12 digits)
 9. CallBackURL Call Back URL URL `https://ip` or `domain:port/path`
 10. AccountReference Account Reference Alpha-Numeric Any combinations of letters and numbers
 11. TransactionDesc Description of the transaction String any string of less than 20 characters
-

- Prompt ussd on customer device:

```
token= token generated from **OAUTH**
push=mpesawrapper.Stkpush().invokeMenu(
    BusinessShortCode="",
    Password="",
    Timestamp="",
    TransactionType= "",
    Amount=,
    PartyA= "",
    PartyB= "",
```

(continues on next page)

(continued from previous page)

```
PhoneNumber= "",
CallBackURL= "",
AccountReference= "",
TransactionDesc= "",
token=token)
```

- check lipa na mpesa status:

```
token= token generated from **OAUTH**
q=mpesawrapper.Stkpush().query_request(BusinessShortCode="", Password="", Timestamp=
↳ "", CheckoutRequestID="", token=token)
```

This api is used to transfer money from one business to another:

```
token=token generated from **OAUTH**
b2b=mpesawrapper.B2B()._b2b_request(
    Initiator = "",
        SecurityCredential="",
        CommandID="",
        SenderIdentifierType="",
        RecieverIdentifierType="",
        Amount="",
        PartyA="",
        PartyB="",
        AccountReference="",
        Remarks="",
        QueueTimeOutURL="",
        ResultURL="",
        token=token
    )
```

Note:

1. CommandID The command id used to carry out a B2B payment String
 - BusinessPayBill
 - BusinessBuyGoods
 - DisburseFundsToBusiness
 - BusinessToBusinessTransfer
 - MerchantToMerchantTransfer
2. Amount The amount been transacted Numeric 1
3. PartyA Organization Sending the transaction Numeric Shortcode (6 digits)

4. **SenderIdIdentifier** Type of organization sending the transaction Numeric 1
 - MSISDN
 - Till Number
 - Organization short code
 5. **PartyB Organization** Receiving the funds Numeric Shortcode (6 digits)
 6. **RecieverIdentifierType** Type of organization receiving the transaction Numeric
 - MSISDN
 - Till Number
 - Organization short code
 7. **Remarks** Comments that are sent along with the transaction. String String of less then 100 characters
 8. **Initiator** This is the credential/username used to authenticate the transaction request.String This is the credential/username used to authenticate the transaction request
 9. **SecurityCredential** This is the encrypted password to autheticate the transaction request String Encrypted password for the initiator to authenticate using the request
 10. **QueueTimeOutURL** The path that stores information of time out transactions.i URL <https://ip> or domain:port/path
 11. **ResultURL** The path that receives results from M-Pesa. URL <https://ip> or domain:port/path
 12. **AccountReference** Account Reference mandatory for “BussinessPaybill” CommandID Alpha-Numeric string of less then 20 characters
-

Transaction Status Request

Check MPESA transaction status:

```
token=token generated from **OAUTH**
a=mpesawrapper.TransactionStatus().get_transaction_status(
    Initiator = "",
    SecurityCredential = "",
    CommandID="",
    TransactionID="",
    PartyA="",
    IdentifierType= "",
    ResultURL="",
    QueueTimeOutURL="",
    Remarks="",
    Occasion="",
    OriginalConversationID="",
    token=token)
```

Note:

1. CommandID Takes only 'TransactionStatusQuery' command id String
 - TransactionStatusQuery
2. PartyA Organization/MSISDN sending the transaction Numeric
 - Shortcode (6 digits)
 - MSISDN (12 Digits)
3. IdentifierType Type of organization receiving the transactionNumeric 1
 - MSISDN
 - Till Number
 - Organization short code
4. Remarks Comments that are sent along with the transaction String sequence of characters up to 100

5. Initiator The name of Initiator to initiating the request Alpha-Numeric This is the credential/username used to authenticate the transaction request
 6. SecurityCredential Encrypted Credential of user getting transaction amount String Encrypted password for the initiator to authenticate the transaction request
 7. QueueTimeOutURL The path that stores information of time out transaction URL <https://ip> or domain:port/path
 8. ResultURL The path that stores information of transaction URL <https://ip> or domain:port/path
 9. TransactionID Unique identifier to identify a transaction on M-Pesa (Use either Transaction ID or Original Conversation ID in the request) Alpha-Numeric LKXXXX1234
 10. OriginalConversationID Unique identifier to identify a request on M-Pesa that has already occurred (Use either Transaction ID or Original Conversation ID in the request) String sXXXX-XXXX-XX
 11. Occasion Optional Parameter String Sequence of characters up to 100
-

Account Balance

Account balance enquiry:

```
token=token generated from **OAUTH**
q=mpesawrapper.Balance().get_balance(
    Initiator="",
        SecurityCredential="",
        CommandID= "",
        PartyA="",
        IdentifierType="",
        Remarks="",
        QueueTimeoutURL="",
        ResultURL="",
        token=token
    )
```

Note:

1. CommandID Takes only 'AccountBalance' CommandID String AccountBalance
2. PartyA Type of organization receiving the transaction Numeric XXXXXX
3. IdentifierType Type of organization receiving the transaction Numeric
 - MSISDN
 - Till Number
 - Organization short code
4. Remarks Comments that are sent along with the transaction. String sequence of characters up to 100
5. Initiator The name of Initiator to initiating the request Alpha-Numeric This is the credential/username used to authenticate the transaction request
6. SecurityCredential Encrypted Credential of user getting transaction amount String Encrypted password for the initiator to authenticate the transaction request

- 7.QueueTimeOutURL The path that stores information of time out transaction URL [https://ip](#) or domain:port/path
- 8.ResultURL The path that stores information of transaction URL [https://ip](#) or domain:port/path
-

Business to customer MPESA transaction:

```
token=generate token from **OAUTH**
b2c=mpesawrapper.B2C().b2c_request(
    InitiatorName="",
    SecurityCredential="",
    CommandID="",
    Amount="",
    PartyA="",
    PartyB="",
    Remarks="",
    QueueTimeOutURL="",
    ResultURL="",
    Occassion="",
    token=token
)
```

Note:

1. InitiatorName The name of the initiator initiating the request Alpha-numeric This is the credential/username used to authenticate the transaction request
2. SecurityCredential Encrypted Credential of user getting transaction amount Alpha-numeric Encrypted password for the initiator to authenticate the transaction request
3. CommandID Unique command for each transaction type
 - SalaryPayment
 - BusinessPayment
 - PromotionPayment
 - Alphanumeric
 - SalaryPayment

- BusinessPayment
 - PromotionPayment
4. Amount The amount been transacted Numbers 00
 5. PartyA Organization /MSISDN sending the transaction Numbers -Shortcode (6 digits) MSISDN (12 digits)
 6. PartyB MSISDN sending the transaction Phone number - Country code (254) without the plus sign -MSISDN (12 digits)
 7. Remarks Comments that are sent along with the transaction. Alpha-numeric sequence of characters upto 100
 8. QueueTimeOutURL The path that stores information of time out transaction URL <https://ip> or domain:port/path
 9. ResultURL The path that stores information of transactions URL <https://ip> or domain:port/path
 10. Occassion Optional Parameter Alpha-numeric sequence of characters up to 100
-

Reversal

API to reverse transactions:

```
token=token generated from **OAUTH**
reversal=mpesawrapper.Reversal().reversal_request(
    Initiator="",
    SecurityCredential="",
    CommandID="TransactionReversal",
    TransactionID="",
    Amount="",
    ReceiverParty="",
    RecieverIdentifierType= "",
    ResultURL="",
    QueueTimeOutURL="",
    Remarks="",
    Occasion="",
    token=token
)
```

Note:

1. CommandID Takes only 'TransactionReversal' Command id String TransactionReversal
2. ReceiverParty Organization /MSISDN sending the transaction Numeric
 - Shortcode (6 digits)
 - MSISDN (12 Digits)
3. ReceiverIdentifierType Type of organization receiving the transaction Numeric 1
 - MSISDN
 - Till Number
 - Organization short code
4. Remarks Comments that are sent along with the transaction. String sequence of characters up to 100

5. Initiator The name of Initiator to initiating the request Alpha-Numeric This is the credential/username used to authenticate the transaction request
 6. SecurityCredential Encrypted Credential of user getting transaction amount String Encrypted password for the initiator to authenticate the transaction request
 7. QueueTimeOutURL The path that stores information of time out transaction URL <https://ip> or domain:port/path
 8. ResultURL The path that stores information of transaction URL <https://ip> or domain:port/path
 9. TransactionID Organization Receiving the funds Alpha-Numeric LKXXXXX1234
 10. Occasion Optional Parameter String sequence of characters up to 100
-

CHAPTER 12

Test Credentials

```
Shortcode 1: 600147
Initiator Name: (Shortcode 1) testapi
Security Credential: (Shortcode 1) Safaricom147!
Shortcode 2: 600000
Test MSISDN: 254708374149
ExpiryDate: 2017-11-13T18:59:13+03:00
Lipa Na Mpesa Online Shortcode: 174379
Lipa Na Mpesa Online PassKey: ↵
↵bfb279f9aa9bdbcf158e97dd71a467cd2e0c893059b10f78e6b72adaled2c919
```