QoS INTEGRATED CENTER INTERFACE SPECIFICATION FOR MOBILE MONEY

Component Name	QoS-IC Mobile Payment Integration Documentation
Document Number	SSL-QS-100
General Description	Detailing exposed interfaces between QoS-IC and Integrating partner applications
Target Audience	Integrating Partners

Created By: Toluwase Ashaolu
Approved By: Nii Boye

Document Scope

This document describes how a third-party service provider will call MobilePayment© Interface from QoS-IC. This implementation addresses connection to a Mobile payment platform through a secure and lighter Interface.

Modification Log

V	*Author*	*Date*	*Summary of Changes*
1.0	Toluwase Ashaolu	10th September,2017	Initial Draft
1.2	Nii Boye	11th September,2017	Approved
2.0	Toluwase Ashaolu	24th September, 2017	Added sections 5, 5.1 and 5.2
3.0	Toluwase Ashaolu	3rd December, 2017	Added sections 6, 6.1 and 6.2
3.1	Toluwase Ashaolu	27th March, 2018	Added sections 10

References

S/N	Document Ref	V	Title

Contents

1. Introduction

1.1 Overview

This document describes how a third party will connect to a MobilePayment© Interface from QoS-IC. Note that all data formats and response definitions are in conformance with the REST standard.

2. Design

2.1 Description

A MobilePayment© transaction provides Mobile Money capability application programming interfaces (APIs) for third-party applications (App for short) to connect to it and use its Mobile Money capability for Bill Payment, Depositing Money in Subscribers account & Querying Subscriber's account in a third-party system.

The subsequent sections show the message structure for Mobile Payment

3. Request Payment

The RequestPayment is a request to debit mobile money from subscribers account.

Partners must code the App based on the API field requirements so that the App can send correct requests accordingly.

3.1 RequestPayment Request (Sent from Third-party)

The following describes the important element required to be sent for RequestPayment request.

S/N	Data Element(DE)	M/O	Description
1.	msisdn	MANDATORY	Subscriber's account i.e. 22967307747
2.	amount	MANDATORY	Transaction amount i.e. 5000
3.	firstname	OPTIONAL	Firstname of the subscriber
4.	lastname	OPTIONAL	Lastname of the subscriber
5.	transref	MANDATORY	Unique transaction reference. To be used as unique identifier in call to request for a payment.
6.	clientid	MANDATORY	A unique generated id for each third-party application. This id is generated by qos and should be included in all payment requests.

Sample Request

```
POST {context}/QosicBridge/user/requestpayment

Content-Type: application/json

[Basic Authentication headers]…
{
  "msisdn": "22967307747",
   "amount": "2000",
   "firstname":"David",
   "lastname":"Ashaolu",
   "transref" :"12345",
   "clientid": "QOS3P001"
 }
```

3.2 RequestPayment Response

The following describes the important data element required for a PaymentRequest Response.

S/N	Data Element(DE)	Description
1.	responsecode	This is the response code for the transaction i.e 00
2.	responsemsg	This describes the response message
3.	transref	This is the transref sent in the request message

Successful Response:

```
HTTP Status Code: 202


{

   "responsecode": "01",

   "responsemsg": "PENDING",

   "transref": “12345”,

   "comment": null

}
```

Un-Successful Response (Failed authentication):

```
HTTP Status Code: 401


HTTP Status 401: Bad credentials
```

Un-Successful Response:

```
HTTP Status Code: 404


{
   "responsecode": "-2",
   "responsemsg": "Invalid Client Id",
   "transref": “12345”,
   "comment": null
}
```

4. Get Transaction Status

A Get Transaction Status message is a request to get the details of an already processed transaction.

4.1 Get Transaction Status Request (Sent from ThirdParty)

The following describes the important element required to be sent for Get Transaction Status request.

S/N	Data Element(DE)	M/O	Description
1.	clientid	MANDATORY	The Unique generated id provided by Qos
2.	transref	MANDATORY	The transaction reference of the payment transaction

Sample Request

```
POST {context}/QosicBridge/user/gettransactionstatus

Content-Type: application/json

[Basic Authentication headers]…
{
   "transref" :"12345",
   "clientid": "QOS3P001"
 }
```

4.2 Get Transaction Status Response

The following describes the important data element required for a Transaction Status Response.

S/N	Data Element(DE)	Description
1.	responsecode	This is the response code for the transaction i.e 00
2.	responsemsg	This describes the response message
3.	transref	This is the transref sent in the request message

Successful Response:

```
HTTP Status Code: 200


{
   "responsecode": "00",

   "responsemsg": "SUCCESSFUL",

   "transref": “12345”,

   "comment": “TRANSACTION APPROVED AND PROCESSED SUCCESSFULLY”

}
```

Un-Successful Response (Failed authentication):

```
HTTP Status Code: 401


HTTP Status 401: Bad credentials
```

Un-Successful Response:

```
HTTP Status Code: 404

{

   "responsecode": "-2",

   "responsemsg": "INVALID CLIENT ID",

   "transref": “12345”,

   "comment": “CLIENT ID DOES NOT EXIST”

}
```

5. Make Deposit

The MakeDeposit is a request to deposit mobile money in Subscriber’s mobile money account. Partners must code the App based on the API field requirements so that the App can send correct requests accordingly.

5.1 MakeDeposit Request (Sent from Third-party)

The following describes the important element required to be sent for MakeDeposit request.

S/N	Data Element(DE)	M/O	Description
1.	msisdn	MANDATORY	Subscriber’s account i.e. 22967307747
2.	amount	MANDATORY	Transaction amount i.e. 5000
3.	transref	MANDATORY	Unique transaction reference. To be used as unique identifier in call to request for a payment.
4.	clientid	MANDATORY	A unique generated id for each third-party application. This id is generated by qos and should be included in all payment requests.

Sample Request

```
POST {context}/QosicBridge/user/deposit

Content-Type: application/json

[Basic Authentication headers]…
{
  "msisdn": "22967307747",
   "amount": "2000",
   "transref" :"56789",
   "clientid": "QOS3P001"
 }
```

5.2 MakeDeposit Response

The following describes the important data element required for a **MakeDeposit** Response.

S/N	Data Element(DE)	Description
1.	responsecode	This is the response code for the transaction i.e 00
2.	responsemsg	This describes the response message
3.	transref	This is the transref sent in the request message

Successful Response:

```
HTTP Status Code: 200

{

   "responsecode": "00",

   "responsemsg": "OPERATION SUCCESSFUL",

   "transref": “56789”,

   "comment": null

}
```

Un-Successful Response (Failed authentication):

```
HTTP Status Code: 401


HTTP Status 401: Bad credentials
```

Un-Successful Response:

```
HTTP Status Code: 404


{
   "responsecode": "-2",
   "responsemsg": "Invalid Client Id",
   "transref": “56789”,
   "comment": null
}
```

6. Refund

The Refund is a request to reverse mobile money back to the Subscriber’s mobile money account. Partners must code the App based on the API field requirements so that the App can send correct requests accordingly.

6.1 Refund Request (Sent from Third-party)

The following describes the important element required to be sent for Refund request.

S/N	Data Element(DE)	M/O	Description
1.	transref	MANDATORY	The initial transaction reference used for RequestPayment. This same transref should be used in call to request for a refund.
2.	clientid	MANDATORY	A unique generated id for each third-party application. This id is generated by QoS and should be included in all payment requests.

Sample Request

```
POST {context}/QosicBridge/user/refund

Content-Type: application/json

[Basic Authentication headers]…
{
   "transref" :"56789",
   "clientid": "QOS3P001"
 }
```

6.2 Refund Response

The following describes the important data element required for a **Refund** Response.

S/N	Data Element(DE)	Description
1.	responsecode	This is the response code for the transaction i.e 00
2.	responsemsg	This describes the response message
3.	transref	This is the transref sent in the request message

Successful Response:

```
HTTP Status Code: 200

{

   "responsecode": "00",

   "responsemsg": "OPERATION SUCCESSFUL",

   "transref": “56789”,

   "comment": null

}
```

Un-Successful Response (Failed authentication):

```
HTTP Status Code: 401


HTTP Status 401: Bad credentials
```

Un-Successful Response:

```
HTTP Status Code: 404

{

   "responsecode": "-2",

   "responsemsg": "Invalid Client Id",

   "transref": “56789”,

   "comment": null

}
```

7. Communication

Requests will be sent over the REST protocol.

8. Security

Requests will be sent over HTTPS only. Each request is also sent using a Basic Authentication. With Basic Authentication, clients send its Base64 encoded credentials **with each request**, using HTTP [Authorization] header. That means each request is independent of other request and server may/does not maintain any state information for the client.

9. Response Code

Please refer to the standard HTTP response codes. Where anything starting with 2XX signifies approved, 4XX means client error and 5XX indicates server error. See Appendix A for response code descriptions.

10. Connecting to QoS Test Platform

Connection to QoS test platform requires login. To request for login credentials, please send email to support@qosic.com and malkiel.adje@qosic.com

APPENDIX A

Sample response Codes obtainable are below. Note that this list is not exhaustive.

S/N	Response Code	Response Description
1.	00	Payment Completed Successfully
2.	01	Payment Awaiting Authorization
3.	-1	Data Error
4.	-2	Invalid CliendID
5.		