**IMOne - Operator Wallet**

**API Web Services**

**Specification Document**

Version <1.0>

**REVISION HISTORY**

**Version Date Remarks Prepared By**

1.0 19 Oct 2017 Release of IMOne – Operator Wallet API document 1.0 Clarence Kok, Widyawati

2

**TABLE OF CONTENTS**

**1. INTEGRATION OVERVIEW..................................................................................................................................... 4**

1.1 DEFINITIONS, ACRONYMS & ABBREVIATIONS .......................................................................................................... 4 1.2 SECURITY ......................................................................................................................................................... 4 1.3 GLOBAL RESPONSES ........................................................................................................................................... 5

**2. REST API (MERCHANT/OPERATOR SIDE).............................................................................................................. 6**

2.1 GET BALANCE ................................................................................................................................................... 6 2.2 PLACE BET ....................................................................................................................................................... 7 2.3 SETTLE BET .................................................................................................................................................... 10 2.4 REFUND/CANCEL ............................................................................................................................................ 13

**APPENDIX.................................................................................................................................................................... 16**

IMONE OPERATOR WALLET – GENERAL BET PLACEMENT AND SETTLEMENT/REFUND FLOW ...................................................... 16

3

**1. Integration Overview**

This integration document will cover the REST API that need to be prepared by Merchant/Operator side in order to perform integration of Operator Wallet Model with IMOne.

For the detail specification of available IMOne APIs for Operator to call (create player, launch game etc), please refer to **IMOne - API Specification vX.X.pdf**.

**1.1 Definitions, Acronyms & Abbreviations**

Term Definition

Merchant/Operator IMOnecustomer that is integrated with IMOne system and

own Players. The term of Merchant and Operator are used interchangeably in this document. Member/Player Customer of the Merchant/Operator and end user of the IM

Lobby and Games. The term of Member and Player are used interchangeably in this document. Provider 3rd party provider of game content that is integrated with

the IMOne system. IM InplayMatrix.

**1.2 Security**

For security reason, operator is expected to setup the environment that restricts the incoming access by IP addresses. IM will provide Operator the IP addresses that will access the Operator side API.

4

**1.3 Global Responses**

The following response codes are being used in all API calls.

Response Code

Respond Message Remarks

0 Successful. Successful response will return this response

code. 612 Invalid Argument. Return this response code if any request

parameter/parameter value is invalid. 999 System has failed to process your

request.

Failed response will return this response code. Operator may append other error details in Response Message.

**Note:**

1. Response must be in JSON format. Refer sample response of each API in the following

sections. 2. Place Bet, Settle Bet and Refund APIs in this specification are idempotent; that is, multiple requests with the same “TransactionId” must have the same effect on your system as if the requests were issued only once. For example, multiple Place Bet requests with the same “TransactionId” must result in money being deducted only once. For successful transactions, the IMOne assumes that any subsequent requests yield the same response.

5

**2. REST API (Merchant/Operator Side)**

This section outlines all APIs need to be prepared at Merchant/Operator Side.

**2.1 Get Balance**

This API returns the current balance of a player.

***Sample URL***: POST http://<domain>/GetBalance

**Sample Input** {

"PlayerId":"myPlayerId”, "Currency": "USD" }

**Property Name**

**Data Type Mandatory Description**

PlayerId String (30) Yes • Player’s ID. This ID is the same ID that operator sent

to IM during player registration. Currency String(3) Yes • Player’s registered Currency with IMOne.

**Response Code**:

Respond Code Respond Message Remark

504 Player does not exist. Return this code if player is not

found at Operator side.

542 Player is inactive. Return this code if player is set to

inactive at Operator side.

**Sample Response (Success):**

{

"Code": 0, "Message": "Successful.", "PlayerID": "myPlayerId", "Currency": "USD", "Balance":189910.50 }

**Sample Response (Failed):**

{

6

"Code": 999, "Message": "System has failed to process your request.", }

**Note:**

1. Response should return Balance at 2 decimal points. 2. Balance must be returned in player registered currency with IM.

**2.2 Place Bet**

This API is called by IM when player place bet. The purpose of the bet request is to initialize start of a game and to deduct the bet amount from the player current account balance.

***Sample URL***: POST http://<domain>/PlaceBet

**Sample Input (Live Dealer)**

{

"ProductWallet": "IMLiveDealer" "Transactions": [

{

"PlayerId":"TF88\_890309", "Provider": "SunWin", "GameId": "imgame13042", "GameName": "VIP Baccarats", "RoundId": "91c780e9-9e4a-e711-80be-0050568c10c1", "BetId": "9afa16ad-c1df-47ae-93f0-b5b942a5200d" "TransactionId": "TLD20170606180048154L8Eg52Na", "Currency": "CNY", "Amount": 10, "Timestamp": "2016-10-10 01:29:33 +08:00" }, {

"PlayerId":"TF88\_890309", "Provider": "SunWin", "GameId": "imgame13042", "GameName": "VIP Baccarats", "RoundId": "91c780e9-9e4a-e711-80be-0050568c10c1", "BetId": "9afa16ad-c1df-47ae-93f0-b5b942a5aaaa", "TransactionId": "TLD20170606180022735gX28sTWH", "Currency": "CNY", "Amount": 20,

7

"Timestamp": "2016-10-10 01:29:33 +08:00" } ] }

**Sample Input (Slot)**

{

"ProductWallet": "IMSlot" "Transactions": [

{

"PlayerId":"TF88\_890309", "Provider": "Intewin", "GameId": "imgameS2201", "GameName": "MonkeyKing", "RoundId": "91c780e9-9e4a-e711-80be-0050568c10c1", "TransactionId": "IMSS111KL00384750432K1W1", "Currency": "CNY", "Amount": 3, "Timestamp": "2017-03-10 01:29:33 +08:00" } ] }

**Property Name**

**Data Type Mandatory Description**

Product Wallet

String(20) Yes • Product Wallet of the bet.

Transactions Array Yes • List of all bet placement transactions.

• Below are attributes of each transaction item. PlayerId String (30) Yes • Player’s ID. This ID is the same ID that operator

sent to IM during player registration. Provider String(50) Yes • Provider Code of the company which provides

the game. GameId String(30) Yes • IM Game Id of the game player places bet. GameName String(200) Yes • IM Game Name of the game player places bet. RoundId String(50) Yes • Round Id of the bet. BetId String(50) Optional • Bet Id of the bet.

• Applicable to **IM Live Dealer** only.

• This is an optional field as some game provider may only provide this during settlement. TransactionId String(50) Yes • IM Transaction Id of the bet. Currency String(3) Yes • Currency of the bet. Amount Float

2 decimals

Yes • Bet amount of the bet.

• Amount should always >= 0.

8

• The amount should be **debited** from the balance. Timestamp Datetime Yes • Timestamp where the transaction received by IM

from provider.

• Timestamp in UTC + 8

• Timestamp format is yyyy-MM-dd HH:mm:ss +08:00

**Response Code**:

Respond Code Respond Message Remark 504 Player does not exist. Return this code if player is not

found at Operator side. 542 Player is inactive. Return this code if player is set to

inactive at Operator side. 510 Insufficient amount. Return this code if player balance is

insufficient to perform the transaction.

**Sample Response (Success):**

{

"Results": [

{

"Code": 0, "Message":"Successful.", "OperatorTransactionId": "trans001", "TransactionId": "TLD20170606180048154L8Eg52Na", "Balance":990.00 }, {

"Code": 0, "Message":"Successful.", "OperatorTransactionId": "trans002", "TransactionId": "TLD20170606180022735gX28sTWH", "Balance":970.00 } ] }

**Sample Response (Failed):**

{

"Code": 510, "Message": "Insufficient amount."

9

}

**Note:**

1. Response should return Balance at 2 decimal points. 2. If any of the transaction item failed, the entire request is considered failed and none of the bet

placement can be accepted. 3. TransactionId is unique identifier across IMOne platform. However, RoundId and BetId may or

may not be unique, subject to game provider behavior.

**2.3 Settle Bet**

This API is called by IM to perform settlement of each bet.

***Sample URL***: POST http://<domain>/SettleBet

**Sample Input (Live Dealer)**

{

"Product Wallet": "IMLiveDealer", "Transactions": [

{

"PlayerId":"TF88\_890309", "Provider": "SunWin", "GameId": "imgame13042", "GameName": "VIP Baccarats", "RoundId": "91c780e9-9e4a-e711-80be-0050568c10c1" "BetId": "9afa16ad-c1df-47ae-93f0-b5b942a5200d", "TransactionId": "TLD20170606175836116oJacw9hS ", "RefTransactionId": ["TLD20170606180048154L8Eg52Na",

"TOQ20170606180048154L8Eaa3321"], "Currency": "CNY", "Amount": 20, "Timestamp": "2016-10-10 02:29:33 +08:00", }, {

"PlayerId":"TF88\_890309", "Provider": "SunWin", "GameId": "imgame13042", "GameName": "VIP Baccarats", "RoundId": "91c780e9-9e4a-e711-80be-0050568c10c1" "BetId": "9afa16ad-c1df-47ae-93f0-b5b942a5aaaa", "TransactionId": "TLD20170606175745328um8icmGl", "RefTransactionId": ["TLD20170606180022735gX28sTWH"], "Currency": "CNY",

10

"Amount ": 0, "Timestamp": "2016-10-10 02:29:33 +08:00", } ] }

**Sample Input (Slot)** {

"Product Wallet": "IMSlot", "Transactions": [

{

"PlayerId":"TF88\_890309", "Provider": "RedStar", "GameId": "imgame13043", "GameName": "Jungle Adventuge", "TransactionId": "TLD20170606175745328um8icmGl", "RefTransactionId": "TLD20170606180022735gX28sTWH", "Currency": "CNY", "Amount ": 0, "Timestamp": "2016-10-10 02:29:33 +08:00" } ] }

**Property Name Data Type Mandatory Description** Product Wallet String(20) Yes • Product wallet of the bet. Transactions Array Yes • List of all bet placement transactions.

• Below are attributes of each transaction item. PlayerId String (30) Yes • Player’s ID. This ID is the same ID that

operator sent to IM during player registration. Provider String(50) Yes • Provider Code of the company which

provides the game. GameId String(30) Yes • IM GameId of the game player places bet. GameName String(200) Yes • IM Game Name of the game player places

bet. RoundId String(50) Yes • Round Id of the bet.

• Applicable to **IM Live Dealer** only. BetId String(50) Yes • Bet Id of the bet.

• Applicable to **IM Live Dealer** only.

• TransactionId String(50) Yes • IM Transaction Id of the bet. RefTransactionId String(50) Yes • Reference IM Transaction Id to be settled.

11

RefTransactionId (Live Dealer)

Array Yes • Reference IM Transaction Id to be settled.

• One request may settle multiple Place Bet Transactions.

• Different request may possess same RefTransactionId. This indicates that the Bet has been split into multiple settlements. Currency String(3) Yes • Currency of the bet. Amount Float

2 Decimals

Yes • Settlement amount of the bet.

• Amount should always >= 0.

• The amount should be **credited** to the balance. Timestamp Datetime Yes • Timestamp where the transaction received

by IM from provider.

• Timestamp in UTC + 8

• Timestamp format is yyyy-MM-dd HH:mm:ss +08:00

**Response Code**:

Respond Code Respond Message Remark 504 Player does not exist. Return this code if player is not

found at Operator side. 545 TransactionId is not found at Operator

side.

Return this code if RefTransactionId is not found at the Operator side.

**Sample Response:**

{

"Results": [

{

"Code": 0, "Message":"Successful", "OperatorTransactionId": "trans003", "TransactionId": "TLD20170606175836116oJacw9hS", "Balance":990.00 }, {

"Code": 999, "Message":"System has failed to process your request.", "TransactionId": "TLD20170606175745328um8icmGl" } ] }

12

**Note:**

1. Response should return Balance at 2 decimal points. 2. If any of the transaction items failed, only the one particular transaction is failed. The rest of

the transactions should still be processed. 3. TransactionId and RefTransactionId is unique identifier across IMOne platform. However,

RoundId and BetId may or may not be unique, subject to game provider behavior.

**2.4 Refund/Cancel**

This API is called by IM to perform refund, in the event of:

i) Bet placement failed. ii) Cancel settlement. iii) Cancel bet placement.

***Sample URL***: POST http://<domain>/Refund

**Sample Input (Live Dealer/Slot)**

{

"Transactions": [

{

"TransactionId": "TLD20170606174802948mcQlR2h9 ", "RefTransactionId": "TLD20170606180048154L8Eg52Na", "TransactionType": "CancelWager" "Timestamp": "2016-10-10 02:29:33 +08:00" }, {

"TransactionId": "TLD20170606174619800SS4MXNx7 ", "RefTransactionId": "TLD20170606180022735gX28sTWH", "TransactionType": "CancelWager", "Timestamp": "2016-10-10 02:29:33 +08:00" } ] }

**Property Name Data Type Mandatory Description** Transactions Array Yes • List of all bet placement transactions.

• Below are attributes of each transaction item. TransactionId String(50) Yes • IM Transaction Id of the bet.

13

RefTransactionId String(50) Yes • Reference IM Transaction Id to be

refunded/cancelled. TransactionType String(50) Yes • There are 2 transaction types: i) CancelWager – Cancel bet

placement/wager ii) CancelSettlement – Cancel settlement

Timestamp Datetime Yes • Timestamp where the transaction received by

IM from provider.

• Timestamp in UTC + 8

• Timestamp format is yyyy-MM-dd HH:mm:ss +08:00

**Response Code**:

Respond Code Respond Message Remark 504 Player does not exist. Return this code if player is not

found at Operator side. 545 TransactionId is not found at Operator

side.

Return this code if RefTransactionId is not found at the Operator side.

**Sample Response:**

{

"Results": [

{

"Code": 0, "Message":"Successful", "OperatorTransactionId": "trans007", "TransactionId": "TLD20170606174802948mcQlR2h9", "Balance":1020.00 }, {

"Code": 999, "Message":"System has failed to process your request.", "TransactionId": "TLD20170606180022735gX28sTWH" } ] }

**Note:**

1. Response should return Balance at 2 decimal points. 2. If any of the transaction items failed, only the one particular transaction is failed. The rest of

14

the transactions should still be processed 3. Please note that Refund/Cancel transactions may cause player balance to become NEGATIVE.

15

**Appendix**

**IMOne Operator Wallet – General Bet Placement and Settlement/Refund Flow**

**Player**

**Game Provider IMOne API Operator API**

Place Bet

Debit Request

Process Data

PlaceBet Request ($)

Success

Success

**Game Logic**

Game Provider process

**Settlement** [parameters] Game Win ($)

Process Data

Settlement Request ($)

Success

Success

**Refund**

[parameters] Game Cancel ($)

Process Data

Refund Request ($)

Success Success

Show Result (Current Balance)

Update status (fire & forget)

Update status (fire & forget)

Update status (fire & forget)

16