POS API Document

1. API Guide
   1. There is test API call I made for testing purpose:

Method: Get

<http://bettwostar.net/api/test>

This will give you following response:

{"rst": "1", "detail": "OK", "data": {"sn": "123456", "datetime": 1512624920}}

1.2 Every request to the API should include following request header:

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Sample** | **When** | **Meaning** |
| sn | "123456" | Always | POS machine serial number.  Unique value to distinguish each logged in user |
| token | "vzpjn1Al43TmssZ1T02mD7Kxp6OZ9BCb6uI6WJC7fzdCwhihxP" | Always except login | POS machine token |

1.3 Every api response is in following structure:

{

"sn": "12345678",

"rst": "1",

"detail": "Success",

"data":

{

"type": "1",

"sort": "2",

…

}

}

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number |
| rst | Result value;//1:success, not 1:fail |
| detail | Message from the server |
| data | Object or Array of data |

1.4 API URL is following type:

{DOMAIN}/api/{API-PATH}

Ex:

http://bettwostar.net/api/test

If rstis not "1", means something went wrong.

And in this case detail will be passed with appropriate message.

1. **POS API**

2.1Login (/login/verify)

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/login/verify

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| Sn | POS machine serial number |
| Password | POS machine unlock password |

Request Data Format:

{

"sn": "SN\_001", //sn: String

"password": "1", //password: String

}

If POS is valid, rst: "1" will be returned.

If not, rst will be something else than "1".

Ex 1:

{

"sn": "123456",

   "rst": "1",

   "token": "vzpjn1Al43TmssZ1T02mD7Kxp6OZ9BCb6uI6WJC7fzdCwhihxP",

   "detail": "Success",

"data":

{

"default\_type": "1",

"default\_sort": "1",

"default\_under": "1"

}

}

Ex 2:

{

"sn": "SN\_001",

"rst": "1003",

"detail": "Your request is failed."

}

--rst ="1"&detail =OK :Success,

-- rst ="1001"&detail =sn\_error:The serial number does not exist,

-- rst ="1002"&detail =pass\_error:wrong password;

-- default\_type:string

//this is the admin set default bet type to this POS machine.

"Nap/Perm":1, "Grouping":2

-- default\_sort:string

//this is the admin set default option to this POS machine.

"100-1":"1", "80-1":"2", "40-1":"3"

-- default\_under:string

//this is the admin set default under to this POS machine.

"U3":"3", "U4":"4", "U5":"5", "U6":"6"

2.2Make bet (/makebet)

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/makebet

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number |
| token | POS machine token |
| data | Object or Array of data |

{

"sn": "SN\_001",

"token": "evD22phd9laHzjO43QLyscJPabBzn4FemBX145BRaHmUGPecaY6heRIAMuvxe2WT",

"data":

[{

"id": "1",

"type": "1",

"sort": "1",

"under":"3",

"amount": "1000",

"games": "1,2,3,4"

},

{

"id":"2",

"type":"2",

"sort":"1",

"under":"3,4",

"amount":"1000",

"games": "1,2,3,4"

},

{

"id":"3",

"type":"3",

"sort":"3",

"under": "6",

"amount":"3000",

"games":

[{

"group": "A",

"number":"2",

"item": "1,2,3,4"

},

{

"group": "B",

"number":"2",

"item": "5,6,7,8"

},

{

"group": "C",

"number":"2",

"item": "9,10,11,12,13"

}]

}]

}

id: POS’s game identification number

type: string, "Nap or Perm":"1", "Combination":"2", "Grouping":"3"

sort: string, "100-1":"1", "80-1": "2", "40-1": "3"

under :string, "U3": "3", "U4": "4", "U5": "5", "U6": "6"

amount: string

apl: string //amount per line = amount/lines

games:

//Nap or Perm and Combination: string

//Grouping: array

group: string, "A","B"…

number: string,

item: string,

If success, rst = "1".

In failure case, error is something else than "1", also detail field is filled up.

Ex 1:

{

"sn": "SN\_001", **//sn is POS’s serial number.**

"rst": "1",

"detail": "Success",

"data": {

"week": "1", //Active week number

"played\_date": "2018-01-08 22:22:38", //user played date(UTC)

"close\_date": "2018-02-05 01:00:00", //bet close time(UTC)

"validity": "2018-02-15", //validity date(UTC)

"total\_stake": "5000", //total stakes amount

"tx\_id": "1", //transaction number: String

"bets": [

{

"id": "1", //POS’s bet identification number

"type": "1",

"sort": "1",

"under": "3",

"amount": "1000",

"games": "1,2,3,4",

"status": "0", //0:right bet, 1:wrong bet

"description": "Success", //bet’s description

"bet\_id": "1321382" //backend bet id (only exist when status is 0.)

"apl": "250"

},

{

"id": "2",

"type": "2",

"sort": "1",

"under": "3,4",

"amount": "1000",

"games": "1,2,3,4",

"status": "0",

"description": "Success",

"bet\_id": "6789526"

"apl": "200"

},

{

"id": "3",

"type": "3",

"sort": "3",

"under": "6",

"amount": "3000",

"games": [

{

"group": "A",

"number": "2",

"item": "1,2,3,4"

},

{

"group": "B",

"number": "2",

"item": "5,6,7,8"

},

{

"group": "C",

"number": "2",

"item": "9,10,11,12,13"

}

],

"status": "0",

"description": "Success",

"bet\_id": "5141092",

"apl": "500"

}

]

}

}

Ex 2:

{

"sn": "SN\_001",

"rst": "1",

"detail": "Success",

"data": {

"week": "1",

"played\_date": "2018-01-08 22:27:36",

"close\_date": "2018-02-05 01:00:00",

"validity": "2018-02-15",

"total\_stake": "1000",

"tx\_id": "1", //transaction number: String

"bets": [

{

"id": "1",

"type": "1",

"sort": "1",

"under": "3",

"amount": "10",

"apl": "2.5",

"games": "1,2,3,4",

"status": "1",

"description": "Your stake amount should be more than 50."

},

{

"id": "2",

"type": "2",

"sort": "1",

"under": "3,4",

"amount": "1000",

"games": "1,2,3,4",

"status": "0",

"description": "Success",

"bet\_id": "6106148",

"apl": "200"

},

{

"id": "3",

"type": "3",

"sort": "3",

"under": "6",

"amount": "60",

"games": [

{

"group": "A",

"number": "2",

"item": "1,2,3,4"

},

{

"group": "B",

"number": "2",

"item": "5,6,7,8"

},

{

"group": "C",

"number": "2",

"item": "9,10,11,12,13"

}

],

"status": "1",

"description": "Your stake must be reached minimum per line requirement."

}

]

}

}

Ex3:

//Game period setting is not set.

{

"sn": "SN\_001",

"rst": "1001",

"detail": "Validy date is not defined."

}

Ex4:

{

"sn": "SN\_001",

"rst": "1003",

"detail": "Your request is failed."

}

week: string

//Active week’s number

status: string ("0" or "1")

//this field reflects bet is wrong or not. When it is wrong status is set to "1", otherwise it is set to "0".

description: string

//when above status value is "0", this is “Success”. Otherwise this value reflects error reason.

Ex: "Your stake must be reached minimum per line requirement.",

"Your balance is 200. It is not enough."

"Game number 100 is not valid."

bet\_id: string

//Random number generated so user or admin can search transactions easily

This value is used for backend. When status value is "0", this field exists. Otherwise this field is not existed.

2.3 POS functions

2.3.1 Result

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/posfunction/result

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number |
| token | POS machine token |
| week | Enter manually(string) |

Request Data Format

{

"sn": "SN\_001", //sn: String

"token": "evD22phd9laHzjOaHmU…", //token: String

"week": "1" //week: String

}

If success, rst = "1".

In failure case, error is something else than 1, also detail field is filled up.

Ex 1:

{

"sn": "123456",

"rst": "1",

"detail": "Success",

"data":

{

"week":"1",

"drawn":["4", "5", "7", "10", "20", "25", "30"],

}

}

Ex 2:

{

"sn": "123456",

"rst": "1003",

"detail": "Your request is failed.",

}

week: String,

//This is the game week number.

drawn: array of string,

//This is the array of string value. Every string value is the number that the admin choose the result number.

\*\*If the result numbers are not drawn, the “drawn” will be empty array.

2.3.2 Reprint

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/posfunction/reprint

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number |
| token | POS machine token |

{

"sn": "123456", //String

"token": "vzpjn1Al43TmssZ1T02mD7Kxp6OZ9BCb6uI6WJC" //String

}

If success, rst = "1".

In failure case, error is something else than 1, also detail field is filled up.

Ex 1:

{

"sn": "123456",

"rst": "1",

"detail": "Success",

"data":

{

"week": "1",

"played\_date": "2018-01-08 22:22:38",

"close\_date": "2018-01-08 22:22:38",

"validity": "2018-02-15",

"total\_stake": "10000",

"tx\_id": "1",

"bets":

[{

"bet\_id": "5496922",

"type":"1",

"sort":"1",

"under":"3",

"amount":"5000",

"games": "1,2,3,7"

},

{

"bet\_id":"5496923",

"type":"3",

"sort":"3",

"under": "4",

"amount":"5000",

"games": [{

"group": "A",

"number":"2",

"item": "1,2,3"

},

{

"group": "B",

"number":"2",

"item": "4,5"

}]

}]

}

Ex2:

{

"sn": "123456",

"rst": "1001",

"detail": "Last transaction does not exist.",

}

Ex3:

{

"sn": "123456",

"rst": "1002",

"detail": "No bet exists",

}

Ex4:

{

"sn": "123456",

"rst": "1003",

"detail": "Your request is failed.",

}

2.3.3 Win List

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/posfunction/winlist

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number |
| token | POS machine token |

{

"sn": "123456",

"token": "vzpjn1Al43TmssZ1T02mD7Kxp6OZ9BCb6uI6WJC",

}

If success, rst = "1".

In failure case, error is something else than 1, also detail field is filled up.

Ex 1:

{

"sn": "123456",

"rst": "1",

"detail": "Success",

"data":

{

"agent": "Fun", //agent’s name

"operator": "Operator1"， //operator’s name

"week": "1",

"close\_date": "2018-02-05 01:00:00",

"winlist":

[{

"bet\_id": "123456",

"amount": "2000"

},

{

"bet\_id": "123126",

"amount": "5000"

}]

"total": "7000",

}

}

Ex 2:

//when there is no winning bet.

{

"sn": "SN\_001",

"rst": "1",

"detail": "Success",

"data": {

"agent": "123",

"operator": "operator1",

"week": "1",

"close\_date": "2018-02-05 01:00:00",

"winlist": [],

"total": "0"

}

}

Ex3:

{

"sn": "123456",

"rst": "1003",

"detail": "Your request is failed.",

}

winlist: object array,

//This is the bet result array.

bet\_id: string,

amount: string,

total: string

//\*\*If the result list are not drawn, the “winlist” will be empty array.

2.3.4 Report

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/posfunction/report

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number |
| token | POS machine token |

{

"sn": "123456",

"token": "vzpjn1Al43TmssZ1T02mD7Kxp6OZ9BCb6uI6WJC",

}

If success, rst = "1".

In failure case, error is something else than 1, also detail field is filled up.

Ex 1:

{

"sn": "123456",

"rst": "1",

"detail": "Success",

"data":

{

"odd\_summary":

[{

"sort ": "1",

"count": "6",

"gross": "13400",

"payable": "9380",

"commission": "4020"

},

{

"sort ": "2",

"count": "6",

"gross": "13400",

"payable": "9380",

"commission": "4020"

}]

"total\_payable": "70235",

"total\_winning": "20000",

"balance\_comp": "50235",

"balance\_agent": "50000"

"status": "Blue"

}

}

Ex 2:

//No bets exist

{

"sn": "SN\_001",

"rst": "1",

"detail": "Success",

"data": {

"odd\_summary": [],

"total\_payable": "0",

"total\_winning": "0",

"balance\_comp": "0",

"balance\_agent": "0",

"status": "Red"

}

}

Ex3:

{

"sn": "123456",

"rst": "1003",

"detail": "Your request is failed.",

}

total\_payable: string,

total\_winning: string,

balance\_comp: string,

balance\_agent: string

status: string

odd\_summary:object array

sort: string

// Odd id 100-1: "1", 80-1: "2", 40-1: "3"

count: string,

gross: string,

payable: string,

commission: string

2.3.5 Credit Limit

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/posfunction/creditlimit

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number |
| token | POS machine token |

{

"sn": "123456",

"token": "vzpjn1Al43TmssZ1T02mD7Kxp6OZ9BCb6uI6WJC",

}

If success, rst = "1".

In failure case, error is something else than "1", also detail field is filled up.

Ex 1:

{

"sn": "123456",

"rst": "1",

"detail": "Success",

"data":

{

"credit\_limit": "900000",

}

}

Ex 2:

{

"sn": "123456",

"rst": "1003",

"detail": "Your request is failed.",

}

credit\_limit: string

//This is POS terminal’s credit limit.

2.3.6 Logout

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/posfunction/logout

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number |
| token | POS machine token |

{

"sn": "123456",

"token": "vzpjn1Al43TmssZ1T02mD7Kxp6OZ9BCb6uI6WJC",

}

If success, rst = "1".

In failure case, error is something else than "1", also detail field is filled up.

Ex 1:

{

"sn": "123456",

"rst": "1",

"detail": "Success",

}

Ex2:

{

"sn": "123456",

"rst": "1003",

"detail": "Your request is failed."

}

2.3.7 Change Password

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/posfunction/changepassword

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number |
| token | POS machine token |
| old\_password | POS machine’s current password |
| new\_password | POS machine’s new password |

{

"sn": "123456",

"token": "vzpjn1Al43TmssZ1T02mD7Kxp6OZ9BCb6uI6WJC",

"old\_password": "abc",

"new\_password ": "ABC",

}

If success, rst = "1".

In failure case, error is something else than "1", also detail field is filled up.

Ex 1:

{

"sn": "123456",

"rst": "1",

"detail": "Success",

}

Ex 2:

{

"sn": "123456",

"rst": "1001",

"detail": "Your credential does not match our record.",

}

Ex 3:

{

"sn": "123456",

"rst": "1003",

"detail": "Your request is failed.",

}

2.3.8 Delete bet

Path: [http://bettwostar.net/api](http://bettwostar.net/api/test)/posfunction/deletebet

Method: POST

|  |  |
| --- | --- |
| Field | Value |
| sn | POS machine serial number(String) |
| token | POS machine token(String) |

{

"sn": "123456",

"token": "vzpjn1Al43TmssZ1T02mD7Kxp6OZ9BCb6uI6WJC",

}

If success, rst = 1.

In failure case, error is something else than 1, also detail field is filled up.

Ex 1:

{

"sn": "123456",

"rst": "1",

"detail": "Success",

}

Ex 2:

{

"sn": "123456",

"rst": "1001",

"detail": "Last transaction does not exist.",

}

Ex 3:

{

"sn": "123456",

"rst": "1002",

"detail": "No bet exists.",

}

Ex 4:

{

"sn": "123456",

"rst": "1003",

"detail": "Your request is failed.",

}

Ex 5:

{

"sn": "123456",

"rst": "1004",

"detail": "This week has closed.",

}