

# API Guide

## Quick Start

### 1、Apply for API Key

```
"agency_uid" : "8dee1e401b87408cca3ca813c2250cb4",  
"aes_key" : "68b074393ec7c5a975856a90bd6fdf47",  
"server_url" : "https://jsgame.live"
```

### 2、Get Game URL (SEAMELESS)

```
Request:  
POST https://{SERVER-URL}/game/v1 HTTP/1.1  
Content-Type: application/json  
{  
  "agency_uid": "8dee1e401b87408cca3ca813c2250cb4", //replace key  
  "timestamp": "1631459081871",  
  "payload": "(AES256EncryptionResult)"  
}  
Response:  
HTTP/1.1 200 OK  
Content-Type: application/json; charset=utf-8  
{  
  "code": 0,  
  "msg": "",  
  "payload": {"game_launch_url": "https://game_url"}  
}
```

### 3、Retrieve Bet Information (SEAMELESS)

Callback to the proxy platform, where the platform implements the interface to receive the information.

```
Request:  
POST https://{callback url} HTTP/1.1  
Content-Type: application/json  
{  
  "agency_uid": "8dee1e401b87408cca3ca813c2250cb4", //replace key
```

```
"payload": "/aIRsjJmK5CYVZLuFS4l5C27vm8Br8..."
"timestamp": "1705474368229"
}
Response:
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
{
  "code": 0
  "msg": "",
  "payload": "m8Br86TSic3fFelsNLxL8eRGXmeuQML.."
}
```

#### 4、Get Game URL (TRANSFER)

```
Request:
POST https://{SERVER-URL}/game/v2 HTTP/1.1
Content-Type: application/json
{
  "agency_uid": "8dee1e401b87408cca3ca813c2250cb4",//replace key
  "timestamp": "1631459081871",
  "payload": "(AES256EncryptionResult)"
}
Response:
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
{
  "code": 0,
  "msg": "",
  "payload": {"game_launch_url": "https://game_url",.....}
}
```

#### 5、Get transaction records

Return the list of transaction records within a specific time period

```
Request:
POST https://{SERVER-URL}/game/transaction/list HTTP/1.1
Content-Type: application/json
{
```

```

"agency_uid": "8dee1e401b87408cca3ca813c2250cb4",//replace key
"timestamp": "1631459081871",
"payload": "(AES256EncryptionResult)"
}
Response:
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
{
  "code": 0,
  "msg": "",
  "payload": {"records":[],"total_page":100,.....}
}

```

## API Interface

### 1、Ge Game URL (SEAMELESS)

The interface is designed to retrieve game URL.

The request URL

<https://{SERVER-URL}/game/v1>

#### Request

URL : https://{SERVER-URL}/game/v1			
Http Method : post			
JSON : <pre>{   "agency_uid": "5d19788698c611ee9b6100163e0b978d",   "timestamp": "1631459081871",   "payload": "(AES256EncryptionResult)" }</pre>			
Parameter	Description	Type	Required
agency_uid	Agent identification code.	String	Yes
timestamp	Current timestamp(Milliseconds)	String	Yes
payload	Encrypt the following JSON String parameters using AES 256 Key: Provided by HUIDU	JSON string	Yes

	For example: Encrypt.AESEncrypt("{\"agency_uid\":\"5d19788698c611ee9b6100163e0b978d\",\"member_account\":\"palyer001\",\"game_uid\":\"1\",\"timestamp\":\"1631459081871\",\"credit_amount\":\"50\"}");		
Parameters within the payload			
timestamp	Current timestamp(Milliseconds)	String	Yes
agency_uid	Game agency identification code	String	Yes
member_account	Player account name(Suggest a username with 4 to 20 characters and a customizable prefix.)	String	Yes
game_uid	Game UID	String	Yes
credit_amount	Game User Credit amount	String	Yes
currency_code	Game currency code (eg. USD)	String	Yes
language	The selected language which the game should be opened. Default: 'en'	String	Yes
home_url	Back to agent website URL	String	No
platform	The platform that the game will open on. Possible values: 1. web (default) 2. H5 (Fill in 1 or 2)	Int	No
callback_url	Game data callback url	String	No

## Response

JSON : { "code": 0, "msg": "", "payload": {"game_launch_url":"game_url"} }		
Parameter	Type	Description
code	Int	Result code
msg	String	Result message

payload	JSON String	Return result
Payload JSON String Parameter correspondence		
game_launch_url	String	Game URL

## 2、Retrieve Bet Information (SEAMELESS)

Callback to the proxy platform, where the platform implements the interface to receive the information.

The request URL

<https://{callback url}>

### Request

URL : <a href="https://{callback url}">https://{callback url}</a>			
Http Method : post			
JSON : <pre>{   "timestamp": "1631459081871",   "payload": "(AES256EncryptionResult)" }</pre>			
Parameter	Description	Type	Required
agency_uid	Agent identification code.	String	Yes
timestamp	The current time is UTC+0. The format is yyyy-MM-dd HH:mm:ss.	String	Yes
payload	Encrypt the following JSON String parameters using AES 256 Key: Provided by HUIDU For example: Encrypt.AESEncrypt("{\"serial_number\":\"245876c4-f717-76d2-b143-3453dac191c7\",\"currency_code\":\"USD\",\"game_uid\":\"83a6890cf84e4c5a6bacf96d5355d472\",\"member_account\":\"play01\",\"win_amount\":\"2\",\"bet_amount\":\"1\",\"timestamp\":\"1631459081871\"}");	JSON String	Yes
Parameters within the payload			
serial_number	A Universally Unique Identifier (UUID) ( <b>The same number is to retry, directly return code=0</b> )	String	Yes

	success and latest balance)		
currency_code	Game currency code (eg. USD)	String	Yes
game_uid	Game UID	String	Yes
member_account	Player account name (recommended username length: 4–20 characters, with a custom prefix).	String	Yes
win_amount	WIN Amount(If the amount is negative, it will be refunded)	String	Yes
bet_amount	BET Amount(If the amount is negative, it will be refunded)	String	Yes
timestamp	Current timestamp(Milliseconds)	String	Yes
game_round	game_round	String	Yes
data	Sports Event Detailed Data	JSON String	Yes

## Response

JSON: <pre>{   "code": 0,   "msg": ""   "payload": "(AES256EncryptionResult)" }</pre>			
Parameter	Description	Type	Required
code	Result Code (0-Success, 1-Failure) Failure will be retried, player balance needs to be returned regardless of success or failure. Returning code=0 and the balance is greater than or equal to 0 means that the bet is successful, and the bet fails in other cases.	Int	Yes
msg	Result message	String	Yes
payload	Encrypt the following JSON String parameters using AES 256 Key: Provided by HUIDU For example: Encrypt.AESEncrypt("{\"credit_am ount\":\"1000.01\",\"timestamp\":\" 1631459081871\"}");	JSON String	Yes

Payload JSON String Parameter correspondence			
credit_amount	Game User Account (credit_amount=credit_amount-bet_amount+win_amount)	String	Yes
timestamp	Current timestamp(Milliseconds)	String	Yes

### 3、Get Game URL (TRANSFER)

This API is used to retrieve the game URL, support deposit and withdrawal transactions, and query transfer information.

The request URL

<https://{SERVER-URL}/game/v2>

#### Request

URL : https://{SERVER-URL}/game/v2			
Http Method : post			
JSON : <pre>{   "agency_uid": "5d19788698c611ee9b6100163e0b978d",   "timestamp": "1631459081871",   "payload": "(AES256EncryptionResult)" }</pre>			
Parameter	Description	Type	Required
agency_uid	Agent identification code.	String	Yes
timestamp	Current timestamp(Milliseconds)	String	Yes
payload	Encrypt the following JSON String parameters using AES 256 Key: Provided by HUIDU For example: Encrypt.AESEncrypt("{\"agency_uid\":\"5d19788698c611ee9b6100163e0b978d\", \"member_account\":\"palyer001\", \"game_uid\":\"1\", \"timestamp\":\"1631459081871\", \"credit_amount\":\"50\", \"transfer_id\":\"1189022\"}");	JSON string	Yes
Parameters within the payload			
timestamp	Current timestamp(Milliseconds)	String	Yes

agency_uid	Game agency identification code	String	Yes
member_account	Player account name (recommended username length: 4–20 characters, with a custom prefix).	String	Yes
game_uid	Game UID	String	No
credit_amount	Transfer amount (greater than 0 for deposit, less than 0 for withdrawal, equal to 0 for query).	String	Yes
currency_code	Currency (see appendix).	String	Yes
language	Language (see appendix).	String	No
home_url	Back to agent website URL	String	No
platform	The platform that the game will open on. Possible values: 1. web (default) 2. H5 (Fill in 1 or 2)	Int	No
transfer_id	Transfer ID is a unique identifier for each transaction.	String	Yes

## Response

JSON: <pre>{   "code": 0,   "msg": "",   "payload": {"game_launch_url": "game_url"} }</pre>			
Parameter	Type	Description	Required
code	Int	Result code	Yes
msg	String	Result message	Yes
payload	JSON String	Return result	Yes
Parameters within the payload			
game_launch_url	String	Game launch url	Yes
player_name	String	Player account name.	Yes
currency	String	Currency (see appendix).	Yes
transfer_amount	String	Transfer amount.	Yes
before_amount	String	Balance before transfer.	Yes



after_amount	String	Balance after transfer.	Yes
transfer_id	String	The operator's transfer ID is a unique identifier for each transaction.	Yes
transaction_id	String	Unique transaction code.	Yes
transfer_status	Int	Transfer status (1 for successful transfer, 2 for failed transfer).	Yes
timestamp	Long	Transfer timestamp.	Yes

## 4、Get transaction records

Return the list of transaction records within a specific time period

The request URL

[{SERVER-URL}/game/transaction/list](#)

### Request

URL: https://{SERVER-URL}/game/transaction/list			
Http Method: post			
JSON: <pre>{   "agency_uid": "5d19788698c611ee9b6100163e0b978d",   "timestamp": "1631459081871",   "payload": "(AES256EncryptionResult)" }</pre>			
Parameter	Description	Type	Required
agency_uid	Agent identification code.	String	Yes
timestamp	Current timestamp(Milliseconds)	String	Yes
payload	Encrypt the following JSON String parameters using AES 256 Key: Provided by HUIDU For example: Encrypt.AESEncrypt("{\"timestamp\": \"1724040409365\", \"agency_uid\": \"5d19788698c611ee9b6100163e0b978d\", \"from_date\": \"172238400000\", \"to_date\": \"1722470399000\", \"page_no\": 1, \"page_size\": 30}");	JSON string	Yes

Parameters within the payload			
timestamp	Current timestamp(Milliseconds))	String	Yes
agency_uid	Agent identification code.	String	Yes
from_date	Start date (UTC+0 timestamp in milliseconds).	Long	Yes
to_date	End date (UTC+0 timestamp in milliseconds).	Long	Yes
page_no	Page number.	Int	Yes
page_size	Page size (minimum: 1, maximum: 5000).	Int	Yes

## Response

JSON: <pre>{   "code": 0,   "msg": "",   "payload": {"records":[{"agency_uid":"abc"...},{...}], "total_page":1000....} }</pre>			
Parameter	Type	Description	Required
code	Int	Result code	Yes
msg	String	Result message	Yes
payload	JSONString	Return result	Yes
Parameters within the payload			
total_count	Int	Total count.	Yes
current_page	Int	Page number.	Yes
page_size	Int	Page size	Yes
records	JSONArray	JSON Array	Yes
Parameters within the records.			
agency_uid	String	Agent identification code.	Yes
member_account	String	Player account name.	Yes
bet_amount	String	BET Amount	Yes
win_amount	String	WIN Amount	Yes
currency_code	String	currency code	Yes
serial_number	String	Unique ID.	Yes

game_round	String	Game round ID.	Yes
game_uid	String	Game UID	Yes
timestamp	String	Transaction time, format UTC+0 (2024-08-19 00:00:00).	Yes

## Appendix :

### AES256:

#### JAVA

##### AES256 ECB Encrypt and Decrypt Example Code

```
import java.security.Security;
import javax.crypto.Cipher;
import javax.crypto.spec.SecretKeySpec;
import org.bouncycastle.jce.provider.BouncyCastleProvider;

public class Encrypt {
    public static boolean initialized = false;
    public static final String ALGORITHM = "AES/ECB/PKCS7Padding";
    public static byte [] Aes256Encode (String str, byte [] key) {
        initialize ();
        byte [] result = null;
        try {
            Cipher cipher = Cipher.getInstance (ALGORITHM, "BC");
            SecretKeySpec keySpec = new SecretKeySpec (key, "AES");
            cipher.init (Cipher.ENCRYPT_MODE, keySpec);
            result = cipher.doFinal (str.getBytes ("UTF-8"));
        } catch (Exception e) e.printStackTrace ();
        return result;
    }
    public static String Aes256Decode (byte [] bytes, byte [] key) {
        initialize ();
        String result = null;
        try {
            Cipher cipher = Cipher.getInstance (ALGORITHM, "BC");
            SecretKeySpec keySpec = new SecretKeySpec (key, "AES");
            cipher.init (Cipher.DECRYPT_MODE, keySpec);
            byte [] decoded = cipher.doFinal (bytes);
            result = new String (decoded, "UTF-8");
        } catch (Exception e) {
```

```

        e.printStackTrace ();
    }
    return result;
}

public static String decrypt(String strToDecrypt,String key){
    try {
        String decryptData = Aes256Decode(Base64.getDecoder().decode(strToDecrypt),
key.getBytes());
        return decryptData;
    }catch (Exception e){
        e.printStackTrace();
    }
    return "";
}

public static String encrypt(String strToEncrypt,String key) {
    return Base64.getEncoder().encodeToString(Aes256Encode(strToEncrypt,
key.getBytes()));
}

public static void initialize () {
    if (initialized) return;
    Security.addProvider (new BouncyCastleProvider ());
    initialized = true;
}
}

```

## PHP

```

<?php

echo base64_encode(openssl_encrypt(
    '{"agency_uid":"03bbfac780fa4df73ade8856358c44b5","timestamp":1711524622000}'
    , 'AES-256-ECB', '055cd814c358c01ae65da946635633ff'
    , OPENSSL_RAW_DATA));

echo openssl_decrypt(base64_decode(
    '6PVSxlZ1nWt98AO4kX/3fhCEaZdHWK/z5RtHpmj9NUAdDyxH/4atn5yVGL1M96DPu
    DVjiTwCJdesQM9bTJsgyKfQpe/AzRM7rkXuESKxUjc='
    )
    , 'AES-256-ECB'

```

```
, '055cd814c358c01ae65da946635633ff'  
, OPENSSL_RAW_DATA);
```

## JS

```
<script  
src="https://cdnjs.cloudflare.com/ajax/libs/crypto-js/4.1.1/crypto-js.min.js"></script>  
<script>  
  
function encrypt() {  
    var key = CryptoJS.enc.Utf8.parse('12345678901234567890123456789012');  
    var plaintext = 'plaintext';  
    var ciphertext = CryptoJS.AES.encrypt(plaintext, key, {  
        mode: CryptoJS.mode.ECB,  
        padding: CryptoJS.pad.Pkcs7  
    });  
    var base64CipherText = ciphertext.toString(); // Base64  
}  
  
function decrypt() {  
    var key = CryptoJS.enc.Utf8.parse('12345678901234567890123456789012');  
    var base64CipherText = 'ciphertext';  
    var decrypted = CryptoJS.AES.decrypt(base64CipherText, key, {  
        mode: CryptoJS.mode.ECB,  
        padding: CryptoJS.pad.Pkcs7  
    });  
    var plaintext = decrypted.toString(CryptoJS.enc.Utf8);  
}  
</script>
```

## API response status.

code	Description
0	Success
10002	Agency not exist
10004	payload error
10005	System error
10008	The game does not exist
10011	Player currencies do not match
10012	Player name already exists, please change player name

10013	Currency is not supported
10014	PlayerName is incorrect
10015	Player account, limited to a-z and 0-9
10016	The account has been frozen. Please contact the administrator
10017	Manufacturer does not exist
10018	This line does not support the current currency
10020	The carrier does not configure a currency
10022	Incorrect parameters
10023	The player name must be at least 3 characters long
10024	Wallet mode does not match
10025	Insufficient wallet balance
10026	Transfer failed
10027	The transfer order already exists
10028	Start and end date cannot be empty
10029	The start and end dates must be the same day
10030	Too many requests, please try again later
10031	Only data within the last 60 days can be queried
10032	End date must be greater than start date
10033	home_url cannot contain ?
10034	System Scheduled Maintenance.

## Game List、Language、Currency

[https://drive.google.com/drive/folders/12FAKtyy2TrWso5CuGAMHoPVWZtwGrCuZ?usp=drive\\_link](https://drive.google.com/drive/folders/12FAKtyy2TrWso5CuGAMHoPVWZtwGrCuZ?usp=drive_link)