

# 吉利 JiLi

## API 规格书

# 修订历程

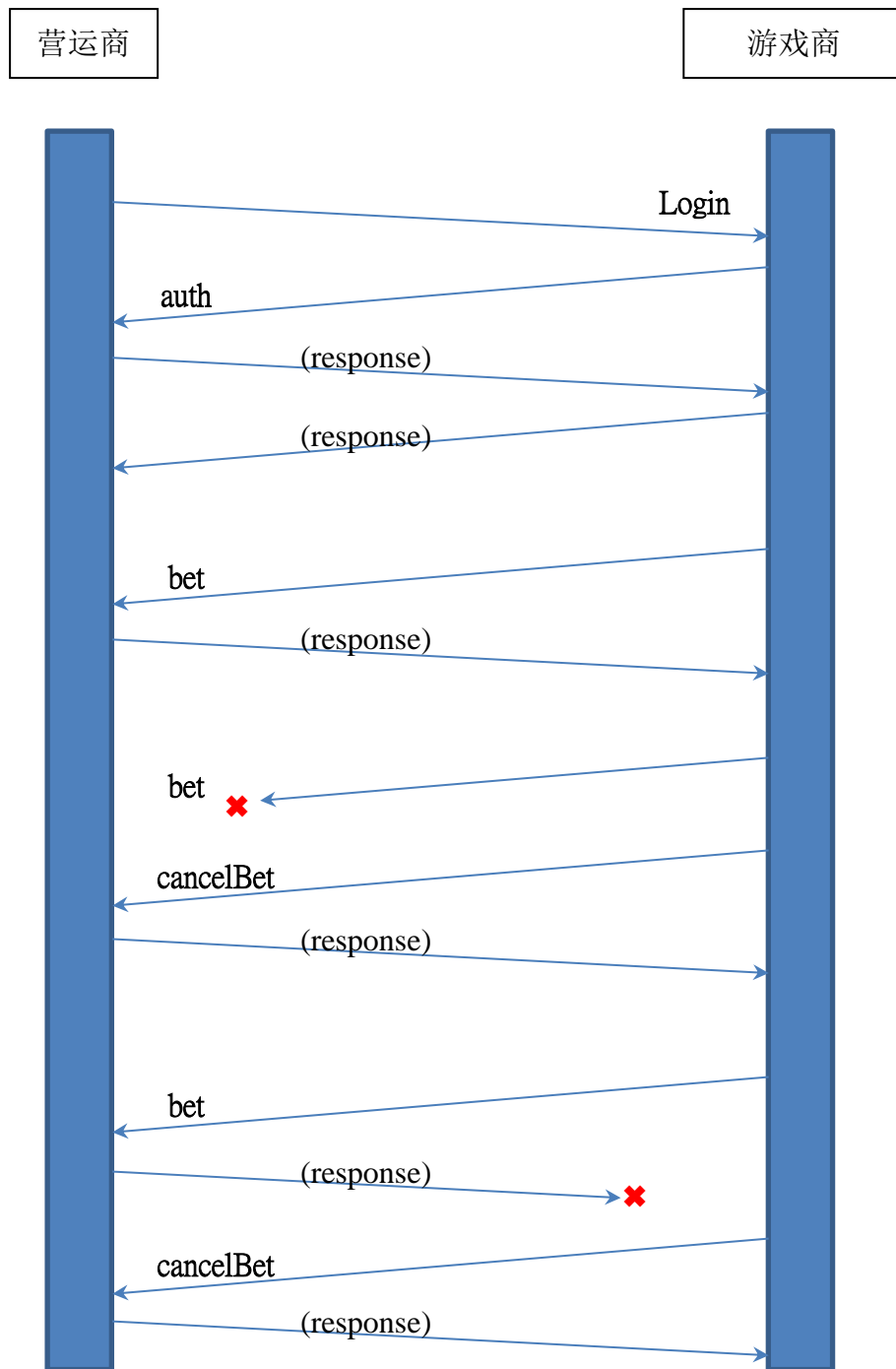
Date	Version	Editor	Change log
2023/7//17	1.0.11	Peter	1. GetBetRecordByUuid (§3.1.5): 弃用
2023/6/28	1.0.10	Peter	新增葡萄牙文及西班牙文 ( <a href="#">附录 A – 语系参数</a> )
2023/6/21	1.0.9	Peter	新增离线模式: 1. sessionBet §4.2.7 2. cancelSessionBet §4.2.8
2023/3/1	1.0.8	Peter	以下增加 token 长度上限 800: 1. Login §2.1.1 2. LoginWithoutRedirect §2.1.2 3. 产生 Token §4.2.1 4. auth §4.2.2
2022/12/9	1.0.7	Peter	GetBetRecordSummary (§3.1.4) 增加 GroupByGame 模式
2022/11/1	1.0.6	Peter	GetGameList (§3.1.6): 返回增加 JP 字段
2022/10/13	1.0.5	Peter	增加 <a href="#">附录 E</a>
2022/9/22	1.0.4	Peter	增加 gameCategory 参数于 Bet §4.2.3 和 SessionBet §4.2.7
2022/9/20	1.0.3	Peter	增加 statementType 参数于 Bet §4.2.3 和 SessionBet §4.2.7
2022/9/14	1.0.2	Peter	增加 sessionTotalBet 参数于 SessionBet §4.2.7
2022/8/25	1.0.1	Peter	以下增加 Platform 参数 1. Login §2.1.1 2. LoginWithoutRedirect §2.1.2 3. Bet §4.2.3 4. SessionBet §4.2.7

## 目录

API 基本流程.....	5
0 设定与注意事项.....	6
1 API 基本通讯及加解密.....	7
1.1.1 Request 与 Response 要求.....	7
1.1.2 Key 产生方式.....	8
1.1.3 Key 产生范例.....	8
2 进入游戏.....	10
2.1.1 登入.....	10
2.1.2 回传登入网址.....	11
3 API.....	12
3.1.1 强制注销游戏.....	12
3.1.2 强制注销游戏(所有玩家或依游戏).....	12
3.1.3 取得注单详细结果连结.....	13
3.1.4 取得注单统计.....	14
3.1.5 查询一笔游戏纪录 (此 API 已弃用).....	16
3.1.6 取得游戏列表.....	17
3.1.7 查询 MustHitBy.....	18
4 由营运商提供 API.....	19
4.1 API 通讯及验证.....	19
4.1.1 通讯.....	19
4.1.2 验证 (非必要).....	19
4.1.3 Response.....	19
4.2 API.....	20
4.2.1 产生 Token.....	20
4.2.2 验证账号.....	21
4.2.3 玩家注单.....	22
4.2.4 特殊需求的注单.....	24
离线开奖.....	24
4.2.5 取消注单.....	25
4.2.6 重送注单.....	28
4.2.7 牌局型注单.....	29
离线模式.....	32
4.2.8 取消牌局型注单.....	33
附录 A – 语系参数.....	35
附录 B – 货币代码.....	35
附录 C – 注单类型.....	36

附录 D - 游戏类型 .....	36
附录 E - HTTP 验证.....	36
HTTP Basic Authentication .....	36

## API 基本流程



# 0 设定与注意事项

- AgentId 又称为站长 Id。
- 不同站别使用 AgentId 用于区分。
- 建立站长时再与我司同步，建立规则可接受英数混合，包含下底线。  
(不建议使用特殊字符)
- 同一 AgentKey 下不允许重复玩家账号名称，即使属于不同 AgentId。建议营运商在玩家账号加上前缀作区别。

举例来说，假如站长 WYZ\_ABET 已经创立/登入 (§2.1.1) 会员 johndoetest123，那么另一个站长 WYZ\_ZBET 就不再允许创立/登入同样名为 johndoetest123 的会员。

- 游戏列表在上线前由我司商务提供，不附在本文件内。
- DB 时区采用 UTC +7 时区。

# 1 API 基本通讯及加解密

## 1.1.1 Request 与 Response 要求

所有请求需要包含以下参数：

Parameter	Type	Description
AgentId	string	代理编号
Key	string	验证参数 (产生方式见 1.2 Key 产生方式)

若使用 HTTP POST 请求，请用 **x-www-form-urlencoded** 来传递参数。

所有响应会包含以下参数：

Parameter	Type	Description
ErrorCode	int	错误代码
Message	string	状态描述
Data	Object	数据数据

游戏商 的返回皆为 JSON 格式。

## 1.1.2 Key 产生方式

Key = {6 个任意字符} + MD5(所有请求参数串 + KeyG) + {6 个任意字符}

- 6 个任意字符：为随机产生，前后不需相同，验证时不看任意字符。
- 所有请求参数串：依照 API 参数列表，按顺序以 key1=value1&key2=value2 格式串起
- 所有 API 的请求参数字符串**最后都要加上 AgentId=xxx**
- 例如：Login (§2.1.1)，表格列出了 Token, GameId, Lang, HomeUrl, Platform  
其中后二项不列入加密计算，因此只需要前三项，按照[表格](#)所列顺序并加上 AgentId，组成以下字符串：

Token=Test1&GameId=101&Lang=zh-CN&AgentId=10081

KeyG = MD5(DateTime.Now.ToString("yyMMd") + AgentId + AgentKey)

- DateTime.Now = **当下 UTC-4 时间**，格式为 yyMMd
  - 年：公元年分末两位
  - 月：两位数，1~9 前面须补 0
  - 日：一位数或两位数，**1~9 前面请不要补 0**
- 举例来说，查询当下时间是 UTC+7 时区 2018/9/30 10:00，换算为 UTC-4 时区为 2018/9/29 23:00，因此应该采用的时间字符串为 180929
- 请注意日期格式：  
例如：2018/2/7 => 18027（7 号是 7 不是 07）  
2018/2/18 => 180218

## 1.1.3 Key 产生范例

- 产生 KeyG

Name	Value
AgentKey	be8e08f95357d215921f91c6a533f74d3194de52
AgentId	abcd_RMB
加密时间	2019/01/01

```
string now = Datetime.Now.ToString("yyMMd");    // 2019/01/01 => "19011"
string agentId = "abcd_RMB";
string agentKey = "be8e08f95357d215921f91c6a533f74d3194de52";
string keyG = MD5(now + agentId + agentKey);
```



KeyG

= MD5("19011abcd\_RMBbe8e08f95357d215921f91c6a533f74d3194de52")

= 9424ffdd6de016a5f90a97a55d99c717

- 以 Login.api 为例

Name	Value
Token	Test006
GameId	1
Lang	zh-CN

```
string keyG = "9424ffdd6de016a5f90a97a55d99c717"
string querystring = "Token=Test006&GameId=1&Lang=zh-CN&AgentId=abcd_RMB";
string md5string = MD5(queryString + keyG);

string randomText1 = "123456";    // Random
string randomText2 = "abcdef";    // Random

string key = randomText1 + md5string + randomText2;
```

md5string

= MD5("Token=Test006&GameId=1&Lang=zh-CN&AgentId= abcd\_RMB9424ffdd6de016a5f90a97a55d99c717")

= 3e4c3b321eac16f20633f683be08d237

Key = "1234563e4c3b321eac16f20633f683be08d237abcdef"

Name	Result
md5string	3e4c3b321eac16f20633f683be08d237
Key	1234563e4c3b321eac16f20633f683be08d237abcdef
LoginUrl	<API URL>/Login?Token=Test006&GameId=1&Lang=zh-CN&AgentId=abcd_RMB &Key=1234563e4c3b321eac16f20633f683be08d237abcdef

## 2 进入游戏

(2.1.1 、 2.1.2 请营运商视需求使用。)

### 2.1.1 登入

URL	<API URL>/singleWallet/Login	Method	GET	Return	HTML
Description	将会员导向进入游戏。				

Request :

Parameter	Type	Require	Description
Token	string	Yes	营运商 api access token (最长 800 字符)
GameId	int	Yes	游戏唯一识别值
Lang	string	Yes	UI 语系, 请参考 <a href="#">附录 A</a>
HomeUrl	string	No	不列入 md5 加密, 游戏回主页功能导向位置
Platform	string	No	不列入 md5 加密, 玩家使用装置, 请参考 §4.2.3, §4.2.7

Error code:

Error Code	Message	Description
2	Error Authority	Key 验证失败 (请参考 §1.1.2)
9	api auth fail	向营运商发送 auth 验证 token 失败
9	Site master not found	AgentId 不存在 (详见 Chapter 0)
9	Site master mismatched	玩家账号与 AgentId 不匹配 (详见 Chapter 0)
104	invalid currency from auth	营运商于 auth 回传错误或禁用的货币

Token 值由营运商产生带入, 用于询问营运商 API (auth, §4.2.2), 进行此次账号登入验证并且取回账号数据, 若取回账号数据为首次都入则会自动建立账号数据

## 2.1.2 回传登入网址

URL	<API URL>/singleWallet/LoginWithoutRedirect	Method	GET	Return	HTML
Description	回传登入网址，不直接导入游戏。				

Request :

Parameter	Type	Require	Description
Token	string	Yes	营运商 api access token (最长 800 字符)
GameId	int	Yes	游戏唯一识别值
Lang	string	Yes	UI 语系, 请参考 <a href="#">附录 A</a>
HomeUrl	string	No	不列入 md5 加密, 游戏回主页功能导向位置
Platform	string	No	不列入 md5 加密, 玩家使用装置, 请参考 §4.2.3, §4.2.7

Response:

Parameter	Type	Require	Description
Data	string	Yes	游戏入口网址  <b>P.S. 网址以 JSON 编码存放于 Data 字段, 须用 JSON 解码撷取出才可使用; 此连结只可使用一次。</b>

Error code: 同 §2.1.1

## 3 API

### 3.1.1 强制注销游戏

URL	<API URL>/api1/KickMember	Method	POST	Return	JSON
Description	强制会员从游戏注销。				

Request:

Parameter	Type	Require	Description
Account	string	Yes	会员唯一识别值

Error code:

Error Code	Message	Description
0	Success	注销成功
101	Invalid Member	会员账号不存在/不在在线

### 3.1.2 强制注销游戏(所有玩家或依游戏)

URL	<API URL>/api1/KickMemberAll	Method	POST	Return	JSON
Description	强制会员从游戏注销。				

Request:

Parameter	Type	Require	Description
GameId	int	No	游戏的唯一识别值

Error code:

Error Code	Message	Description
0	Success	注销成功

QueryString 范例:

//不传参数: 强制注销所有会员

AgentId=abcd\_RMB

//传 GameId: 强制注销在该游戏内的所有会员

GameId=1&AgentId=abcd\_RMB

### 3.1.3 取得注单详细结果连结

URL	<API URL>/api1/GetGameDetailUrl	Method	POST	Return	JSON
Description	取得一局游戏详细信息的 URL				

Request:

Parameter	Type	Require	Description
WagersId	bigint	Yes	在游戏内注单唯一值
Lang	string	NO	指定语系 <b>P.S. 计算 key 时请忽视</b>

Response:

Parameter	Type	Require	Description
Url	string	Yes	游戏详细结果 Url

Error code:

Error Code	Message	Description
0	Success	查询成功
101	Record Not Exist	游戏纪录不存在

### 3.1.4 取得注单统计

URL	<API URL>/GetBetRecordSummary	Method	Get	Return	JSON
Description	取得 时间范围 内一款游戏注单统计数值				

Request:

Parameter	Type	Require	Description
StartTime	DateTime	Yes	查询开始时间
EndTime	DateTime	Yes	查询结束时间
GameId	int	No	游戏的唯一识别值 未带时, 回传结果为所有游戏加总 <b>P.S. 计算 key 时请忽视</b>
Currency	string	No	查询货币代码 未带时, 回传结果所有币别都会个别加总 <b>P.S. 计算 key 时请忽视</b>
FilterAgent	bool	No	只查询所带之代理 (AgentId) 下的注单统计 未带时, 回传结果为所有代理加总 <b>P.S. 计算 key 时请忽视</b>
GroupByGame	int	No	查询结果依照游戏分开统计 0 或不带 = 不启用 1 = 启用 <b>P.S. 计算 key 时请忽视</b>

Response:

Parameter	Type	Require	Description
BetAmount	decimal(16,4)	Yes	投注总金额
PayoffAmount	decimal(16,4)	Yes	派彩总金额
WagersCount	int	Yes	注单数量
Currency	string	Yes	注单币别
GameId	int	No	游戏的唯一识别值 只使用于 GroupByGame=1 时

Error code:

Error Code	Message	Description
0	Success	查询成功
2	Invalid Key	检查码错误
101	Invalid DateTime	不正确的时间格式, 或开始时间大于结束时间

4	(other failures)	其他错误
---	------------------	------

Sample：时间格式请参考下列

```
StartTime = '2020-09-02T00:00:00';
EndTime = '2020-09-02T23:59:59';
```

回传值范例:

1. 所有游戏加总统计 (GroupByGame = 0)

```
{
  "ErrorCode": 0,
  "Message": "",
  "Data": [
    {
      "BetAmount": 3002.5,
      "PayoffAmount": 1.95,
      "Turnover": 3002.5,
      "Preserve": 0,
      "WagersCount": 6,
      "Currency": "THB",
    }
  ]
}
```

2. 各游戏个别统计 (GroupByGame = 1)

```
{
  "ErrorCode": 0,
  "Message": "",
  "Data": [
    {
      "BetAmount": 2.5,
      "PayoffAmount": 1.95,
      "Turnover": 2.5,
      "Preserve": 0,
      "WagersCount": 5,
      "Currency": "THB",
      "GameId": 108
    },
  ],
}
```

```
{
  "BetAmount": 3000,
  "PayoffAmount": 0,
  "Turnover": 3000,
  "Preserve": 0,
  "WagersCount": 1,
  "Currency": "THB",
  "GameId": 182
}
]
```

### 3.1.5 查询一笔游戏纪录 (此 API 已棄用)

URL	<API URL>/GetBetRecordByUuid	Method	Get	Return	JSON
Description	以 游戏纪录唯一值 查询一笔游戏纪录				

Request:

Parameter	Type	Require	Description
WagersId	bigint	Yes	在游戏内注单唯一值

Response:

Parameter	Type	Require	Description
Account	string	Yes	会员唯一识别值
WagersId	bigint	Yes	在游戏内注单唯一值
GameId	int	Yes	游戏的唯一识别值
WagersTime	DateTime	Yes	投注时间
BetAmount	decimal(16,4)	Yes	投注金额
PayoffTime	DateTime	No	派彩时间
PayoffAmount	decimal(16,4)	No	派彩金额
Status	int	Yes	注单状态:— 1: 赢 2: 输
SettlementTime	DateTime	Yes	对帐时间
GameCategoryId	int	Yes	游戏类型:— 请参考章节 3.1.6
Type	Int	Yes	注单类型:— 1: main game 9: free game 11: 道具卡
AgentId	String	Yes	会员所属站长唯一识别值

Error code:

Error Code	Message	Description
0	Success	查询成功
101	Record Not Exist	游戏纪录不存在



## 3.1.6 取得游戏列表

URL	<API URL>/GetGameList	Method	POST	Return	JSON
Description	查询所有游戏的 GameId、名称、类型等信息				

Response:

Parameter	Type	Require	Description
GameId	string	Yes	游戏的唯一识别值
name	object	Yes	zh-CN: 简体中文名称 zh-TW: 繁体中文名称 en-US: 英文名称
GameCategoryId	int	Yes	游戏类型: 请参考 <a href="#">附录 D</a>
JP	boolean	Yes	true: 此款游戏内建 JP false: 无内建 JP

## 3.1.7 查询 MustHitBy

URL	<API URL> /GetMustHitBy	Method	Get	Return	JSON
Description	以 游戏 ID 查询取得满额必开信息				

Request:

Parameter	Type	Require	Description
GameId	Int	No	游戏唯一识别值。不带值则会返回全部游戏的满额必开信息。 <b>P.S. 计算 key 时请忽略 GameId</b>

Response:

Parameter	Type	Require	Description
Currency	String	Yes	币别名称
Game	String	Yes	游戏名称
GameId	Int	Yes	游戏 ID 识别值
MustHitByPool	Decimal	Yes	满额必开水箱
MustHitByValue	Decimal	Yes	满额必开数值

Error code:

Error Code	Message	Description
0	Success	查询成功
9	Execution Error	后端执行错误
101	Record Not Exist	纪录不存在

## 4 由营运商提供 API

营运商至少需要提供 auth, bet, cancelBet 三支 API, 其他的可与游戏商确认是否有需求。

### 4.1 API 通讯及验证

须提供信息

- `<api_url>` 营运商 API 接口位置

#### 4.1.1 通讯

所有请求接使用 HTTP POST

参数传递回传使用 JSON 格式

HTTP request header:

`Content-type: application/json`

#### 4.1.2 验证 (非必要)

營運商可要求遊戲商在請求的標頭中加入驗證碼。

支援的驗證方式請參考[附錄 E](#)。

這項驗證必非必要。

#### 4.1.3 Response

营运商的 response 应为 JSON 格式, 并包含以下共通字段:

Parameter	Type	Require	Description
errorCode	int	Yes	查询成功与否的状态代码, 0 表示成功, 其他值请参考各 API
message	string	No	查询失败时的错误讯息, 查询成功或不需要时可省略

## 4.2 API

### 4.2.1 产生 Token

- 营运商须帮会员产生及保存 token, 并在呼叫 Login (§2.1.1) 或 LoginWithoutRedirect (§2.1.2) 时带给游戏商。
- 游戏商向营运商发送的验证和投注会携带此 token。
- 营运商需要能用 token 来辨认会员。
- Token 长度上限为 800 字符

提示: 营运商可考虑采用 UUID, MD5 或是任何可以产生不重复字符串的算法来产生 token。Token 有效期限由营运商自行决定。

## 4.2.2 验证账号

URL	<API URL>/auth	Method	POST	Return	JSON
Description	验证是否为合法玩家				

Request:

Parameter	Type	Require	Description
reqId	string	Yes	請求的識別唯一代碼，用於後續追查 API 呼叫狀況
token	string	Yes	营运商 api access token (最长 800 字符)

Response:

Parameter	Type	Require	Description
username	string	Yes	账号唯一识别名称
currency	string	Yes	身上持有货币种名称, 请参考 <a href="#">附录 B</a>
balance	double	Yes	持有货币量
token	string	No	营运商 api access token 若有更新需要以此字段回传

Error code:

Error Code	Message	Description
0	Success	查询成功
4	Token expired	Api access token 已过期或无效
5	Other error	其他错误 (详情请放到 message 字段)

范例 **Post Body**

```
{
  "reqId": "0af0c835-c37b-5da0-9e4e-25463e6ed14d"
  "token": "6f6d63331c1173c8367e43b5fe6c49dd"
}
```

响应范例

```
{
  "errorCode": 0,
  "message": "success",
  "username": "testUser",
  "currency": "USD",
  "balance": 1000,
  "token": "6f6d63331c1173c8367e43b5fe6c49dd"
}
```

## 4.2.3 玩家注单

本节所述 bet 用于老虎机及鱼机。关于棋牌游戏及押分机游戏，请参考 § 4.2.7

URL	<API URL>/bet	Method	POST	Return	JSON
Description	传送一笔玩家注单给营运商。 若因为网络或其他因素而未获得营运商的响应，游戏商可以： 1. 取消该注单 (请见 § 4.2.5); 2. 或重送该注单 (请见 §4.2.6)。 营运商与游戏商应事先约定好要采取哪一种行动。 P.S. 部分游戏有离线后才开奖，此种注单 isFreeRound 为 true，并带 userId 识别玩家身份，详见 §4.2.4。				

Request:

Parameter	Type	Require	Description
reqId	string	Yes	请求标识符。同一注单重送请求时会与前次不同。(长度最长 50)
token	string	Yes	营运商 api access token
currency	string	Yes	货币名称
game	int	Yes	游戏代码
round (*)	bigint	Yes	注单唯一识别值 (等同 §3.1.3 WagersId)
wagersTime	bigint	Yes	注单结账时间戳
betAmount	double	Yes	押注金额
winloseAmount	double	Yes	派彩金额
isFreeRound	bool	No	为 true 时表示此注单为离线开奖 (请参考 § 4.2.4)
userId	string	No	玩家账号唯一值 (依营运商需求带入; <a href="#">离线开奖</a> 一定会带)
transactionId	bigint	No	1. 鱼机游戏大单号 (依营运商需求带入) 或 2. <a href="#">离线开奖</a> 的触发局局号
platform	string	No	玩家装置信息 (依营运商需求带入; 营运商须在 Login § 2.1.1 /LoginWithoutRedirect § 2.1.2 带入)
statementType	int	No	注单类型 (依营运商需求带入; 请参考 <a href="#">附录 C</a> )
gameCategory	int	No	游戏类型 (依营运商需求带入; 请参考 <a href="#">附录 D</a> )

\* round 参数是玩家注单的唯一识别值。取消注单也是以此识别。

Response:

Parameter	Type	Require	Description
username	string	Yes	账号唯一识别名称
currency	string	Yes	身上持有货币种名称
balance	double	Yes	持有货币量
txId	bigint	No	营运商承认注单后提供的交易识别唯一值 (注单已承认也请附上)
token	string	No	营运商 api access token 若有更新需要以此字段回传

Error code:

Error Code	Message	Description
0	Success	查询成功
1	Already accepted	该注单已承认
2	Not enough balance	玩家余额不足
3	Invalid parameter	参数无效 (详情请放到 message 字段)
4	Token expired	Api access token 已过期或无效
5	Other error	其他错误 (详情请放到 message 字段)

## 范例 Post Body

```
{
  "reqId": "9177b749-cf37-585b-b17c-cfd5024ca6e2",
  "token": "6f6d63331c1173c8367e43b5fe6c49dd",
  "currency": "USD",
  "game": 1,
  "round": 17238050501001102002,
  "wagersTime": 1592559162073,
  "betAmount": 10,
  "winloseAmount": 5
}
```

响应范例

```
{
  "errorCode": 0,
  "message": "success",
  "username": "testUser",
  "currency": "USD",
  "balance": 995,
  "txId": 108430126,
  "token": "6f6d63331c1173c8367e43b5fe6c49dd"
}
```

## 4.2.4 特殊需求的注单

特殊注單是帶有特定欄位的玩家注單 (§4.2.3), 同樣使用 `<API URL>/bet`。

### 离线开奖

相关字段: `isFreeRound`, `userId`, `transactionId`

部份开奖可能会在玩家离线后才发生。由于玩家离线时没有 `token`, 因此这类的注单会使用该玩家最近一次登入时的 `token`, 并在参数中加上 `userId` 来识别玩家身分, 同时再增加 `isFreeRound` 参数以便与一般的注单区别。此外, `transactionId` 参数表示该离线开奖的触发局。

以下是目前可离线的开奖:

- 舞龙争霸 (game ID = 9) 的免费游戏
- 梅杜莎 (game ID = 101) 的免费游戏

**P.S.** 离线开奖注单如果逾时或遇到其他错误, 将会持续重发直到成功或尝试 10 次。如营运商不希望重发, 须告知游戏商设定为不重发, 则逾时或错误时会像普通注单一样发送取消。

#### 范例 **Post Body**

```
{
  "reqId": "9177b749-cf37-585b-b17c-cfd5024ca6e2",
  "token": "6f6d63331c1173c8367e43b5fe6c49dd",
  "currency": "USD",
  "game": 9,
  "round": 17238050501001102002,
  "wagersTime": 1592559162073,
  "betAmount": 0,
  "winloseAmount": 55,
  "userId": "3368799_2679638",
  "isFreeRound": true,
  "transactionId": 1630891368000155009
}
```



## 4.2.5 取消注单

本节所述 cancelBet 用于老虎机及鱼机。关于棋牌游戏及押分机游戏，请参考 § 4.2.8

URL	<API URL>/cancelBet	Method	POST	Return	JSON
Description	取消一笔注单。 游戏商应持续发送直到获得营运商回应。此项请求可于玩家脱机时发送, token 可能已经失效, 因此除了玩家最后一次使用的 token 之外还会带上 userId, 营运商可自行选择使用 userId 或 token 来识别玩家身份。				

Request:

Parameter	Type	Require	Description
reqId	string	Yes	请求标识符。同一取消重送请求时会与前次不同。(长度最长 50)
currency	string	Yes	货币名称
game	int	Yes	游戏代码
round	bigint	Yes	注单唯一识别值 (等同 § 4.2.3 round)
betAmount	double	Yes	押注金额
winloseAmount	double	Yes	派彩金额
userId	string	Yes	玩家账号唯一值
token	string	Yes	该注单发生时的 token

\* round 参数是玩家注单的唯一识别值。取消注单也是以此识别。

Response:

Parameter	Type	Require	Description
username	string	Yes	账号唯一识别名称
currency	string	Yes	身上持有货币种名称
balance	double	Yes	持有货币量
txId	bigint	No	营运商取消注单后提供的交易识别唯一值 (注单已取消也请附上)

Error code:

Error Code	Message	Description
0	Success	取消成功
1	Already canceled	该注单已取消
2	Round not found	注单无效
3	Invalid parameter	参数无效 (详情请放到 message 字段)
5	Other error	其他错误 (详情请放到 message 字段)
6	Already accepted and cannot be canceled	注单已成立而不可取消

## 请求范例 Post Body

```
{
  "reqId": "9177b749-cf37-585b-b17c-cfd5024ca6e2",
  "currency": "USD",
  "game": 1,
  "round": 17238050501001102002,
  "betAmount": 10,
  "winloseAmount": 5,
  "userId": "3368799_2679638",
  "token": "6f6d63331c1173c8367e43b5fe6c49dd",
}
```

## 返回范例

```
{
  "errorCode": 0,
  "message": "success",
  "username": "testUser",
  "currency": "USD",
  "balance": 1000,
  "txId": 108430126
}
```

## 范例 Post Body

```
{
  "reqId": "9177b749-cf37-585b-b17c-cfd5024ca6e2",
  "currency": "USD",
  "game": 1,
  "round": 17238050501001102002,
  "betAmount": 10,
  "winloseAmount": 5,
  "userId": "3368799_2679638",
  "token": "6f6d63331c1173c8367e43b5fe6c49dd",
}
```

## 响应范例

```
{
  "errorCode": 0,
  "message": "success",
  "username": "testUser",
  "currency": "USD",
  "balance": 1000,
  "txId": 108430126
}
```

## 4.2.6 重送注单

许多原因都可能导致注单未能正确获得营运商的响应，例如网络等待时间导致逾时或线路不良而的遗失封包。如果营运商允许，游戏商可重送注单。

重送注单依照以下规则进行：

1. 如果仍未获得回应，游戏商会再尝试重送。连同第一次发送，同一笔注单最多发送 3 次。此次数可依营运商需求调整；
2. 若 3 次都未获得回应，则依照 §4.2.5 向营运商取消该笔注单；
3. 当营运商收到一笔已经认可的注单时，响应中的 error code 可以是 0 也可以是 1，前者表示成功而后者表示该注单已经认可过（请参考 §4.2.3）；响应中仍需要有该玩家所持有的余额。

### 响应范例

```
{
  "errorCode": 1,
  "message": "already accepted",
  "username": "testUser",
  "currency": "USD",
  "balance": 995,
  "txId": 108430126,
  "token": "6f6d63331c1173c8367e43b5fe6c49dd"
}
```

## 4.2.7 牌局型注单

URL	<API URL>/sessionBet	Method	POST	Return	JSON
Description	<p>传送一笔牌局型玩家注单给营运商，用于有开牌结算的游戏。分成下注及结算两类型。</p> <p>若因为网络或其他因素而未获得营运商的响应，游戏商会采取以下行为：</p> <ol style="list-style-type: none"> <li>1. 结算前，取消当笔下注并中止牌局，然后进行结算；</li> <li>2. 结算后会将派彩发送给营运商，若不成功须持续发送；达到 n 次不成功（预设 n=10）时，游戏商应立即通知营运商进行处理。</li> </ol> <p>取消牌局型注单请参考 §4.2.8。</p> <p>P.S. 由于结算有可能发生在玩家离线之后，因此牌局型注单会有参数 <code>userId</code>，营运商可自行选择用 <code>token</code> 或 <code>userId</code> 识别玩家身分。另一种方式是使用<a href="#">离线模式</a>。</p>				

Request:

Parameter	Type	Require	Description
reqId	string	Yes	请求标识符。同一请求重送请求时会与前次不同。（长度最长 50）
token	string	Yes	营运商 api access token
currency	string	Yes	货币名称
game	int	Yes	游戏代码
round (*)	bigint	Yes	下注/结算行为唯一识别值
offline	boolean	No	请参考 <a href="#">离线模式</a>
wagersTime	bigint	Yes	时间戳（依 type 不同为下注时间或结账时间）
betAmount	double	Yes	押注金额
winloseAmount	double	Yes	派彩金额
sessionId (**)	bigint	Yes	牌局唯一值（所有下注及结算共享同一值，等同 §3.1.3 WagersId）
type	int	Yes	注单类型： 1=下注 2=结算
userId	string	Yes	玩家账号唯一值
turnover	double	Yes	有效投注金额，结算才会带入 ※注 1
preserve	double	No	预扣金额（仅用于有预先扣款机制的游戏）※注 2
platform	string	No	玩家装置资讯（依营运商需求带入；营运商须在 Login §2.1.1 /LoginWithoutRedirect §2.1.2 带入）
sessionTotalBet	double	No	牌局全部下注总和（依营运商需求带入；只在结算时带入）

statementType	int	No	注单类型 (依营运商需求带入; 请参考 <a href="#">附录 C</a> )
---------------	-----	----	--

\* round 参数是玩家注单的唯一识别值。取消下注也是以此识别。

\*\* sessionId 是整局注单的唯一识别值。所有的下注/结算会有相同的 sessionId。

Response:

Parameter	Type	Require	Description
username	string	Yes	账号唯一识别名称
currency	string	Yes	身上持有货币种名称
balance	double	Yes	持有货币量
txId	bigint	No	营运商承认注单后提供的交易识别唯一值 (注单已承认也请附上)
token	string	No	营运商 api access token 若有更新需要以此字段回传

Error code:

Error Code	Message	Description
0	Success	注单成功
1	Already accepted	该注单已承认
2	Not enough balance	玩家余额不足
3	Invalid parameter	参数无效 (详情请放到 message 字段)
4	Token expired	Api access token 已过期或无效
5	Other error	其他错误 (详情请放到 message 字段)

※注 1

有效投注额 (*turnover*) 在大多数游戏中等同于押注金额 (*betAmount*)。不过有部分游戏可以双边下注, 为防止玩家利用双边下注来洗返水, 这类的游戏会提供不同于押注金额的有效投注金额, 计算方式如下:

$$turnover = \min(betAmount, |winloseAmount - betAmount|)$$

※注 2

有部分游戏 (例如 Rummy) 在结算时才会算出实际的押注金额 (*betAmount*), 为了防止玩家从营运商提走所有余额使结算失败, 这类游戏会带 *preserve* 作为保证金, 营运商须在下注阶段扣取保证金, 并在结算阶段退还给玩家。范例如下:

下注:

```
{"reqId":"e4034252-86ca-40dc-9e29-8a9beb24f28c","token":"ae97a7179a4bc47a99c753704a340f82","currency":"THB","game":94,"round":1654662770005413094,"wagersTime":1655192382,"betAmount":0,"winloseAmount":0,"sessionId":1654662770005303094,"type":1,"turnover":0,"preserve":12800}
```

结算:

```
{"reqId":"63b3e8ba-53fe-48e6-b2c3-ac218b89a1c7","token":"ae97a7179a4bc47a99c753704a340f82","currency":"THB","game":94,"round":1654662770005513094,"wagersTime":1655192471,"betAmount":912,"winloseAmount":18240,"sessionId":1654662770005303094,"type":2,"turnover":912,"preserve":12800}
```

以下公式同时适用于有预先扣款机制以及无该机制的游戏（无预扣可视为  $preserve = 0$ ）

下注：下注后余额 = 下注前余额  $- betAmount - preserve$

结算：结算后余额 = 结算前余额  $- betAmount + preserve + winloseAmount$

## 离线模式

牌局型注单的结算 (/sessionBet with type=2) 或取消 (/cancelSession) 送出时所带的 token 可能已过期。营运商可要求游戏商使用离线模式。离线模式默认并不启用。此模式下会带 offline 参数 (其值恒为 true) 及离线 token。营运商需要先给予游戏商离线 token 密钥, 游戏商藉此用以下公式产生离线 token:

SHA224(<离线 token 密钥> + <round id> + <session id> + '\_' + <user id>)

举例来说, 当离线 token 密钥为 AAAA-BBBB-CCCC-DDDD, 若 round=26727840008124608, sessionId=26727838908124090, userId=APLAYER 以离线模式发送结算, 则 token 为:

SHA224 (AAAA-BBBB-CCCC-DDDD2672784000812460826727838908124090\_APLAYER) =  
1cb22d550f2d7e755631435c28b9a08b08519f49f6fba46095f755b6

游戏商将发送以下请求给营运商:

```
{
  "reqId": "0d8098b1-1ee2-47cb-b159-9149e2fb945b",
  "token": "1cb22d550f2d7e755631435c28b9a08b08519f49f6fba46095f755b6",
  "currency": "USD",
  "game": 124,
  "round": 26727840008124608,
  "offline": true,
  "wagersTime": 1687348800,
  "betAmount": 0,
  "winloseAmount": 0,
  "sessionId": 26727838908124090,
  "type": 2,
  "turnover": 60
}
```

请注意离线模式下 offline 的值为 true 且不会带 userId。



## 4.2.8 取消牌局型注单

URL	<API URL>/cancelSessionBet	Method	POST	Return	JSON
Description	取消一笔牌局型注单。 游戏商应持续发送直到获得营运商回应。此项请求可于玩家脱机时发送，token 可能已经失效，因此除了玩家最后一次使用的 token 之外还会带上 userId，营运商可自行选择使用 userId 或 token 来识别玩家身分。				

Request:

Parameter	Type	Require	Description
reqId	string	Yes	请求标识符。同一请求重送时会与前次不同。(长度最长 50)
currency	string	Yes	货币名称
game	int	Yes	游戏代码
round (*)	bigint	Yes	下注请求唯一识别值 (等同 § 4.2.7 round)
offline	boolean	No	请参考 <a href="#">离线模式</a>
betAmount	double	Yes	押注金额 type=1 为该笔投注金额,
winloseAmount	double	Yes	派彩金额。此请求中恒为 0。
userId	string	Yes*	玩家账号唯一值
token	string	Yes	该注单发生时的 token 或 <a href="#">离线模式</a> 的 token
sessionId	bigint	Yes	牌局唯一识别值 (等同 § 4.2.7 sessionId)
type	int	Yes	注单类型: 1=下注
preserve	double	No	预扣金额 (等同 § 4.2.7 preserve)

\* round 参数是下注/结算的唯一识别值。取消下注也是以此识别。

\* [离线模式](#)并不会带 userId

Response:

Parameter	Type	Require	Description
username	string	Yes	账号唯一识别名称
currency	string	Yes	身上持有货币种名称
balance	double	Yes	持有货币量
txId	bigint	No	营运商取消注单后提供的交易识别唯一值 (注单已取消也请附上)

Error code:

Error Code	Message	Description
0	Success	取消成功

1	Already canceled	该注单已取消
2	Round not found	注单无效
3	Invalid parameter	参数无效 (详情请放到 message 字段)
5	Other error	其他错误 (详情请放到 message 字段)

说明: 牌局型注单的特性是, 必须等前一笔下注成功了才会进行下一笔, 下注都成功后会进行结算, 或是某一笔下注失败后中止牌局也进行结算。

- 取消下注 (type=1): 成功时只会取消单笔下注, 同局的其他下注不受影响。正常情况下, 只会取消最后一笔下注, 而且必须是还未结算的牌局。
  - 接受取消后, 这一局就不可再接受任何下注, 只会有结算。之后游戏商对同一局发送的下注无效。
  - 有可能因为网络等待时间或其他因素, 导致营运商在收到下注前就收到取消, 对营运商来说是取消一笔不存在的下注, 这时营运商应注记该局为失败, 之后同一局的下注均无效。
  - 正常情况游戏商不会在结算后发送 type 1 取消。但有可能因为网络等待时间或其他因素, 导致营运商在结算后才收到 type 1 的取消。若营运商收到此种 type 1 取消, **应取消该笔下注**, 原因是游戏商是以该笔下注失败为前提进行结算。
- 游戏商不会对结算 (type=2) 发送取消。

## 附录 A – 语系参数

语系	代码
简中	zh-CN
繁中	zh-TW
英语	en-US
泰文	th-TH
印度尼西亚	id-ID
越南	vi-VN
印度	hi-IN
坦米尔文	ta-IN
日文	jp-JP
韩文	ko-KR
葡萄牙文	pt-BR
西班牙文	es-AR

## 附录 B – 货币代码

代码	代表货币
CNY	人民币
USD	美金
THB	泰铢
MMK	缅甸元
INR	印度卢比
IDR	印尼盾
MYR	马来西亚林吉特
VND	越南盾
kVND	K 越南盾
kIDR	K 印尼盾
JPY	日圆
BND	文莱元
SGD	新加坡元
HKD	港元
PHP	菲律宾比索
RUB	俄罗斯卢布
BDT	孟加拉塔卡

LKR	斯里兰卡卢比
BRL	巴西雷亚尔
MXN	墨西哥比索
PKR	巴基斯坦卢比
kKHR	K 柬埔寨瑞尔
kLAK	K 老挝基普
KRW	韩圆
EUR	欧元

## 附录 C – 注单类型

代码	注单类型
1	一般下注
9	离线开奖 (请参考 §4.2.4)
11	道具卡
12	购买 bonus

## 附录 D – 游戏类型

代码	游戏类型
1	电子
2	棋牌
3	游戏大厅
5	捕鱼
8	押分机 (含宾果)

## 附录 E – HTTP 验证

本节列出目前可用的 HTTP 验证。

### HTTP Basic Authentication

HTTP Basic Authentication 是由 username 与 password 组成并作 Base64 运算得到简易认证码, 请参考[维基百科](#)获取更多信息; username 和 password 由营运商提供。

范例如下:

- username:abc, password:abc123
- abc:abc123
- Base64 encode: YWJjOmFiYzEyMw==
- HTTP request header:

```
Authorization: Basic YWJjOmFiYzEyMw==
```

以下为一个带有 http basic authentication 的完整 auth 请求 (§ 4.2.2):

Body:

```
Authorization: Basic YWJjOmFiYzEyMw==
```

```
Content-Type: application/json
```

Body:

```
{"reqId": "011d3624-90cd-477b-bf05-b8b2c623ee45", "token": "dc7b7f79187bbc17cc2994965a736a9c"}
```