

目录

1. 文件历史记录	2
2. API 概述.....	2
3. API 登录.....	4
3.1. 访问令牌登录	4
3.2. 正常登录.....	5
4. APIs	6
4.1. 打开游戏 URL	6
4.2. 游戏列表.....	6
4.3. 游戏回合状态	7
4.4. 游戏历史 URL	8
4.5. 余额	9
4.6. 提现	9
4.7. 验证提现/充值事务	10
4.8. 按日期获取提现/充值事务	11
4.9. 登出	12
5. 锦标赛 APIs	13
5.1. 开放锦标赛大厅流程.....	13
5.2. 获取锦标赛列表.....	13
5.3. 获取锦标赛信息.....	16
5.4. 获取锦标赛等级.....	18
6. 运营商集成 APIs.....	20
6.1. 验证令牌.....	20
6.2. 验证用户名/密码（可选）	20
6.3. 余额	21
6.4. 下注	21
6.5. 下注结算.....	22
6.6. 取消下注.....	23
6.7. 赢取奖金.....	23

6.8. 赢取头奖.....	24
6.9. 事务	25
6.10. 提现	25
6.11. 充值	26
7. 运营商集成锦标赛 API	27
7.1. 加入锦标赛	27
7.2. 取消加入锦标赛.....	28
7.3. 锦标赛 Win	29
8. 参考	30
8.1. 时间戳	30
8.2. 散列	30
8.3. 状态代码.....	31
9. 常见问题	33

1. 文件历史记录

日期	版本	说明	修订人
2021-Jul-07	1.23	添加了 5. 锦标赛 APIs 添加了 7. 运营商集成锦标赛 API	Customer Support
2021-May-05	1.21	添加了 4.9. 登出 API	Customer Support
2021-Feb-18	1.20	ID, RoundId: 按成员唯一, 仅对新客户有效	Customer Support
2020-Nov-6	1.19	将用户名长度扩展为 32 个字符	Customer Support
2020-June-30	1.17	获取 Withdraw/Deposit 事务	Customer Support
2020-April-20	1.15	删除了 extendedinfo	Customer Support
2020-April-08	1.14	添加了 bet API 的 extendedinfo 参数	Customer Support
2020-Mar-27	1.13	用户名: 4~20 个字母数字字符, 不区分大小写	Customer Support
2020-Feb-12	1.12	添加了 settle-bet API 的 extendedinfo 参数	Customer Support
2019-July-24	1.10	添加了 balance, withdraw API	Customer Support
2019-March-22	1.0	创建	Customer Support

2. API 概述

在文件中使用了通用术语:

- **提供者**：是指 **API/System**。
- **运营商**：是指需要与我们的系统集成的人员。在本文件中，运营商和运营商系统是同义词。

本文件介绍了运营商与无缝钱包 API 集成而需要实现的一组 API。

提供者 API 和运营商集成 API 使用**散列**验证传入请求的完整性。散列加密算法的描述见第 8.2 节。

有些 API 需要具备**幂等性**，这意味着发出多个相同请求与发出单个请求的效果相同。

运营商集成 API 响应指示请求是否成功的状态整数。状态参考的描述见第 8.3 节。

APIs 超时为 2.5 秒。

3. API 登录

提供者支持 2 种登录方法：

- 访问令牌登录（适用于网站）
- 正常登录（适合于移动应用程序）

3.1.访问令牌登录

当运营商在其大厅中托管提供者的游戏时，可以使用访问令牌登录。流程如下：

1. 玩家登录运营商系统后，运营商系统传递**令牌**以打开游戏链接：
`{gaming-url}/playGame?token={Token}&appID={AppID}&gameCode={GameCode}&language={language}&mobile={isMobile}&redirectUrl={redirectUrl}`
2. 游戏客户端调用提供者 API，后者调用运营商集成 API，对**令牌**进行身份验证。
3. 运营商集成 API 响应以验证令牌、用户名和余额。
4. 提供者游戏网站显示游戏。

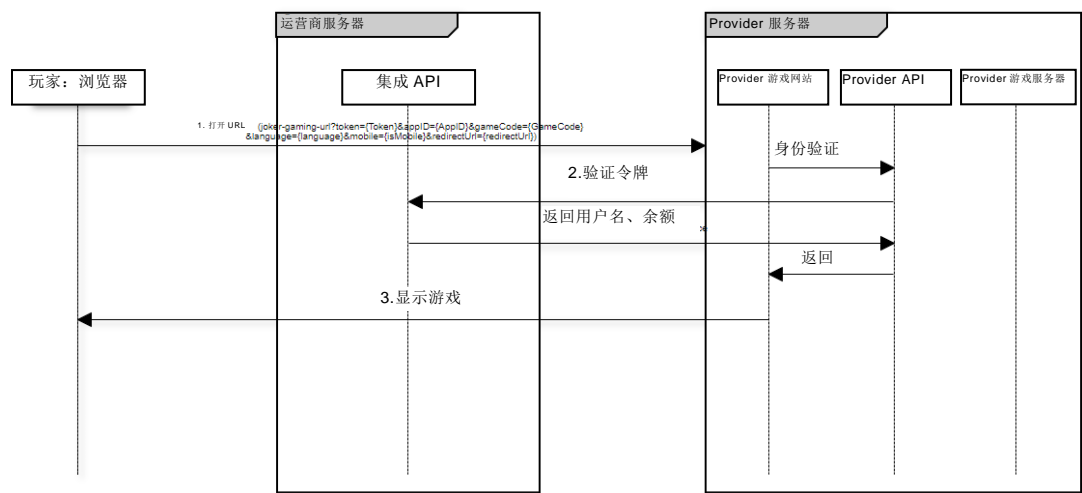


图 1：访问令牌登录

3.2. 正常登录

我们提供正常登录选项，允许运营商的玩家直接登录游戏平台（移动应用程序）。

1. 玩家下载游戏平台（移动应用程序）并使用用户名/密码登录。
2. 游戏客户端调用 API，后者调用运营商身份验证集成 API 来验证用户名/密码。
3. 运营商集成 API 响应以验证用户名/密码以及令牌和余额。
4. 游戏应用程序显示游戏。

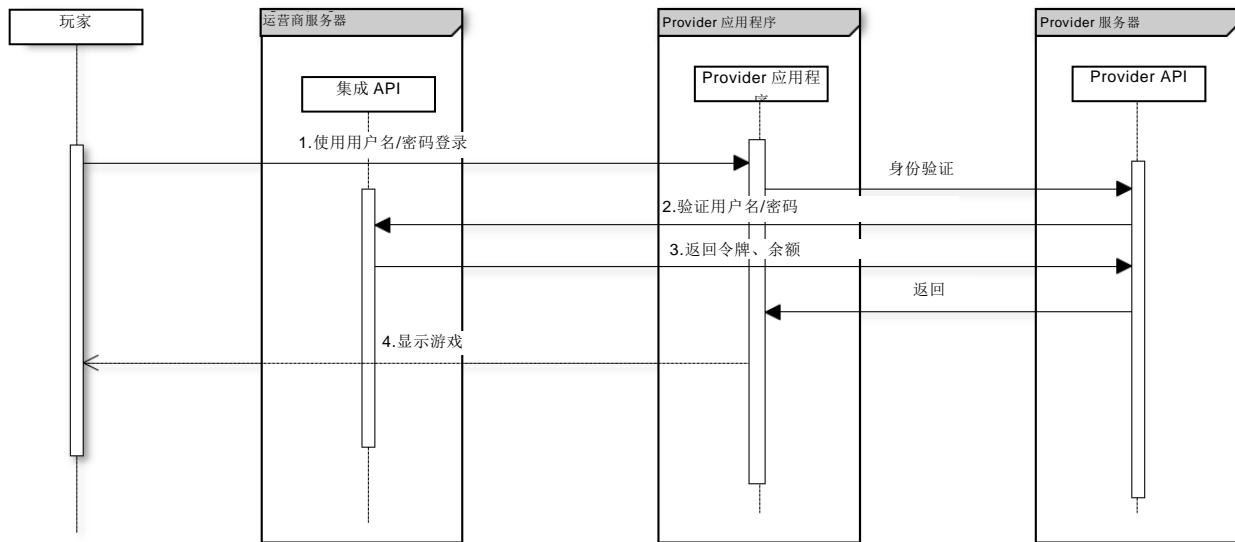


图 2：正常登录

4. APIs

4.1. 打开游戏 URL

请求

```
GET {gaming-url}/playGame?token={token}&appID={appID}&gameCode={gameCode}&language={language}&mobile={isMobile}&redirectUrl={redirectUrl}
```

属性	类型	必填	说明
appID	String	Y	
token	String	Y	由运营商生成。玩家由令牌唯一标识。
gameCode	String	Y	提供者列出的游戏代码。
language	String	N	玩家的首选语言 (ISO 639-1, 2 letter code)
mobile	Boolean	N	true/false
redirectUrl	String	N	当客户端注销时，URL 被重定向

4.2. 游戏列表

请求

```
POST {api-url}/list-games
Content-Type: application/json
```

```
{"AppID":"AppID","Hash":"792d2414e3b0c77e9ddea9e8aa924364","Timestamp":1528210236}
```

属性	类型	必填	说明
AppID	String	Y	
Hash	String	Y	
Timestamp	Long	Y	UNIX 时间戳

响应

```
{
  "Error": "0",
  "Description": "OK",
  "ListGames": [
    {
      "GameType": "Slot",
      "GameCode": "dxxsh3dfmjpio",
      "GameName": "Tai Shang Lao Jun",
      "SupportedPlatforms": "Desktop,Mobile",
```

```

    "Specials": "hot,new",
    "Order": 1,
    "DefaultWidth": 960,
    "DefaultHeight": 630,
    "Image1": "//sampledomain/gameimages/landscape/fwria11mjbrwh.png"
  }
  ....
]
}

```

属性	类型	必填	说明
Error	String	Y	成功 = “0” 失败 = “1”
Description	String	Y	

数组:

属性	类型	必填	说明
GameCode	String	Y	游戏标识符
GameName	String	Y	游戏名称
DefaultWidth	Integer	Y	游戏窗口宽度
DefaultHeight	Integer	Y	游戏窗口高度
GameType	String	Y	游戏类型, 例如 Slot、Fishing
SupportedPlatForms	String	Y	
Specials	String	Y	
Order	Integer	Y	
Image1	String	Y	

4.3.游戏回合状态

请求

POST {api-url}/game-round-status
Content-Type: application/json

```

{"AppID":"AppID","RoundID":"uhax9gb5upqgq","Username":"DEMO","Hash":"792d2414e3b0c77e9ddea9e8aa924364","Timestamp":1528210236}

```

属性	类型	必填	说明
AppID	String	Y	
Hash	String	Y	
RoundID	String	Y	

Timestamp	Long	Y	UNIX 时间戳
Username	String	Y	不要在哈希中使用此属性

响应: Content-Type: application/json

```
{
  "Error": "0",
  "Description": "OK",
  "Bets": [
    {
      "ID": "gh6gqyfdyxnx6",
      "Status": "Settled",
      "Amount": 0.10,
      "Result": 0.6
    }
  ]
}
```

数组:

属性	类型	必填	说明
ID	String	Y	下注 ID
Status	String	Y	成功、 已取消、 已结算

4.4. 游戏历史 URL

请求

POST {api-url}/game-history-url

Content-Type: application/json

```
{"AppID": "AppID", "Username": "DEMO", "GameCode": "xea1rt9gebhwy", "RoundID": "uhax9gb5upqgq", "Language": "en", "Hash": "792d2414e3b0c77e9ddea9e8aa924364", "Timestamp": 1528210236}
```

响应: Content-Type: application/json

```
{
  "Error": "0",
  "Description": "OK",
  "Url":
    "//History?Signature=k2Nyoas94qHkAaoIKuYAdJEnr9c%3d&Key=xkqryq3b5aeuq&Type=Game&Timestamp=1552962025093&Lang=en"
}
```


属性	类型	必填	说明
URL	String	Y	

4.5. 余额

API返回玩家的当前信用余额

请求

POST {api-url}/balance
Content-Type: application/json

```
{"AppID":"AppID","Username":"DEMO","Hash":"792d2414e3b0c77e9ddea9e8aa924364","Timestamp":1528210236}
```

响应: Content-Type: application/json

```
{
  "Error": "0",
  "Description": "OK",
  "Amount": 100.01
}
```

属性	类型	值	必填	说明
Amount	Decimal	Positive numeric	Y	

4.6. 提现

API允许运营商触发充值API（充值API用于将资金从提供者的系统返回到运营商的系统）

请求

POST {api-url}/withdraw
Content-Type: application/json

```
{"AppID":"AppID","Username":"DEMO","Hash":"792d2414e3b0c77e9ddea9e8aa924364","Timestamp":1528210236}
```

响应: Content-Type: application/json

```
{
  "Error": "0",
  "Description": "OK"
}
```

4.7. 验证提现/充值事务

请求

POST {api-url}/ statement-by-request

Content-Type: application/json

```
{
  "Username": "DEMO",
  "AppID": "AppID",
  "IDs": " 5ee99dcba3787, 5ee98746d9216",
  "Timestamp": "1593144363603",
  "Hash": "abb7cc87ea3914e46c3a9562dfe89832"
}
```

属性	类型	必填	说明
AppID	String	Y	
Hash	String	Y	
Username	String	Y	
Timestamp	Long	Y	UNIX 时间戳
IDs	String	Y	列表请求 ID 用逗号分隔

响应: Content-Type: application/json

```
{
  "Error": "0",
  "Description": "OK",
  "Data": [
    {
      "Id": "5ee99dcba3787",
      "Username": "AppID.DEMO",
      "Time": "2020-06-24T15:38:55.169",
      "Amount": 20.00,
      "Type": "TopUp"
    },
    {
      "Id": "5ee98746d9216",
      "Username": "AppID.DEMO ",
      "Time": "2020-06-24T15:40:04.152",
      "Amount": -3594.90,
      "Type": "Withdraw"
    }
    .....
  ]
}
```

属性	类型	必填	说明
Error	String	Y	
Description	String	Y	
Data	Array	Y	

Id	String	Y	
Username	String	Y	
Time	String	Y	
Amount	Decimal	Y	
Type	String	Y	<ul style="list-style-type: none"> - TopUp: 充值 - Withdraw: 提现

4.8.按日期获取提现/充值事务

请求

POST {api-url}/ statement-by-date

Content-Type: application/json

```
{
  "Username": "DEMO",
  "AppID": "AppID",
  "Date": "2020-06-24",
  "Timestamp": "1593498325050",
  "Hash": "edc6c90c32b1e148d9c2bbf4f4eeb311"
}
```

属性	类型	必填	说明
AppID	String	Y	
Hash	String	Y	
Username	String	Y	
Timestamp	Long	Y	UNIX 时间戳
Date	String	Y	日期格式: yyyy-MM-dd

响应: Content-Type: application/json

```
{
  "Error": "0",
  "Description": "OK",
  "Data": [
    {
      "Id": "5ee99dcba3787",
      "Username": "AppID.DEMO",
      "Time": "2020-06-24T15:38:55.169",
      "Amount": 20.00,
      "Type": "TopUp"
    },
    {
      "Id": "5ee98746d9216",
      "Username": "AppID.DEMO",
      "Time": "2020-06-24T15:40:04.152",
      "Amount": -3594.90,
      "Type": "Withdraw"
    },
    .....
  ]
}
```

属性	类型	必填	说明
Error	String	Y	
Description	String	Y	
Data	Array	Y	
属性	类型	必填	说明
Id	String	Y	
Username	String	Y	
Time	String	Y	
Amount	Decimal	Y	
Type	String	Y	<ul style="list-style-type: none"> - TopUp: 充值 - Withdraw: 提现

4.9. 登出

请求

POST {api-url}/sign-out
Content-Type: application/json

```
{"AppID":"AppID","Username":"Username","Hash":"792d2414e3b0c77e9ddea9e8aa924364","Time stamp":1620184560896}
```

属性	类型	必填	说明
AppID	String	Y	
Username	String	Y	
Hash	String	Y	
Timestamp	Long	Y	UNIX 时间戳

响应

```
{
  "Error": "0",
  "Description": "Sign out successfully."
}
```

属性	类型	必填	说明
Error	String	Y	成功 = “0” 失败 = “1”
Description	String	Y	

5. 锦标赛 APIs

5.1. 开放锦标赛大厅流程

请求

```
GET {gaming-url}/playGame?token={token}&appID={appID}&gameCode=Tournament&language={language}&mobile={isMobile}&redirectUrl={redirectUrl}
```

5.2. 获取锦标赛列表

请求

```
POST {api-url}/tournaments
Content-Type: application/json
```

```
{"AppID":"AppID","Status":"All","Timestamp":"1625640447662","Hash":"155786eaa5211c74b45ab371c97b86ca"}
```

属性	类型	必填	说明
AppID	String	Y	
Status	String	Y	状态具有如下值： - 全部 - 活跃 - 完成 - 全新
Hash	String	Y	
Timestamp	Long	Y	UNIX 时间戳

响应

```
{
  "Error": "0",
  "Description": "OK",
  "Data": [
    {
      "TournamentID": "xeahz7p7oag51",
      "TournamentName": "TaiShangLaoJunGW",
      "StartDate": "2021-07-07T20:00:00",
      "EndDate": "2021-07-08T22:00:00",
      "Description": "Tai Shang Lao Jun",
      "Status": "New",
      "CurrencyCode": "MYR",
      "Prize": [
        {

```

```
    "PrizeOCode": "nknnny55cpitgg",
    "Amount": 10000,
    "Qty": 1,
    "Position": 1,
    "Ticket": 1,
    "Points": 1000000
  },
  {
    "PrizeOCode": "nknnny55cpitgg",
    "Amount": 5000,
    "Qty": 2,
    "Position": 2,
    "Ticket": 1,
    "Points": 500000
  },
  {
    "PrizeOCode": "nknnny55cpitgg",
    "Amount": 1000,
    "Qty": 10,
    "Position": 4,
    "Ticket": 1,
    "Points": 100000
  }
],
"Cost": 1125,
"RequiredTicket": 0,
"ImageLink": "http://dl.changxingwnet.com/tournament/assets/icon/248x248/5/TaiShangLaoJunGW.png",
"ImageLinkSmall": "http://dl.changxingwnet.com/tournament/assets/web-
icon/320x265/1/TaiShangLaoJunGW.png",
"Type": "total_win",
"GameCode": "dxxsh3dfmjpio",
"PrizePool": 39000,
"NoOfSpins": 250,
"TypeDescription": "Total Win",
"TypeName": "Total Win",
"PoolContribution": 0
},
.....
]
```

属性	类型	必填	说明
Error	String	Y	成功 = “0” 失败 = “1”
Description	String	Y	
Data	Array	Y	
属性	类型	必填	说明
TournamentID	String	Y	
TournamentName	String	Y	
StartDate	DateTime	Y	
EndDate	DateTime	Y	
Description	String	Y	
Status	String	Y	
CurrencyCode	String	Y	
Cost	Decimal	Y	
RequiredTicket	Integer	Y	
ImageLink	String	Y	
ImageLinkSmall	String	Y	
Type	String	Y	
GameCode	String	Y	
PrizePool	Decimal	Y	
NoOfSpins	Integer	Y	
TypeDescription	String	Y	
TypeName	String	Y	
PoolContribution	Decimal	Y	
Prize	Array	Y	
属性	类型	必填	说明
PrizeOCode	String	Y	
Amount	Decimal	Y	
Qty	Integer	Y	
Position	Integer	Y	
Ticket	Integer	Y	
Points	Long	Y	

5.3. 获取锦标赛信息

请求

POST {api-url}/tournament
Content-Type: application/json

```
{"AppID": "AppID", "TournamentID": "xeahz7p7oag51", "Timestamp": "1625640447662", "Hash": "1936d0c41de7d522501b6b028f4fa178"}
```

属性	类型	必填	说明
AppID	String	Y	
TournamentID	String	Y	
Hash	String	Y	
Timestamp	Long	Y	UNIX 时间戳

响应

```
{
  "Error": "0",
  "Description": "OK",
  "Data": {
    "TournamentID": "xeahz7p7oag51",
    "TournamentName": "TaiShangLaoJunGW",
    "StartDate": "2021-07-07T20:00:00",
    "EndDate": "2021-07-08T22:00:00",
    "Description": "Tai Shang Lao Jun",
    "Status": "New",
    "CurrencyCode": "MYR",
    "Prize": [
      {
        "PrizeOCode": "nknnny55cpitgg",
        "Amount": 10000.0,
        "Qty": 1,
        "Position": 1,
        "Ticket": 1,
        "Points": 1000000
      },
      {
        "PrizeOCode": "nknnny55cpitgg",
        "Amount": 5000.0,
        "Qty": 2,
        "Position": 2,
        "Ticket": 1,
        "Points": 500000
      }
    ]
  }
}
```



```

{
  "PrizeOCode": "nknnny55cpitgg",
  "Amount": 3000.0,
  "Qty": 3,
  "Position": 3,
  "Ticket": 1,
  "Points": 300000
},
{
  "PrizeOCode": "nknnny55cpitgg",
  "Amount": 1000.0,
  "Qty": 10,
  "Position": 4,
  "Ticket": 1,
  "Points": 100000
}
],
"Cost": 1125.00,
"RequiredTicket": 0,
"ImageLink":
"http://dl.changxingwnet.com/tournament/assets/icon/248x248/5/TaiShangLaoJunGW.png",
"ImageLinkSmall": "http://dl.changxingwnet.com/tournament/assets/web-
icon/320x265/1/TaiShangLaoJunGW.png",
"Type": "total_win",
"GameCode": "dxxsh3dfmjpio",
"PrizePool": 39000.0,
"NoOfSpins": 250,
"TypeDescription": "Total Win",
"TypeName": "Total Win",
"PoolContribution": 0.0
}
}

```

属性	类型	必填	说明
Error	String	Y	成功 = “0” 失败 = “1”
Description	String	Y	
Data	Object	Y	
属性	类型	必填	说明
TournamentID	String	Y	
TournamentName	String	Y	
StartDate	DateTime	Y	
EndDate	DateTime	Y	
Description	String	Y	
Status	String	Y	

CurrencyCode	String	Y	
Cost	Decimal	Y	
RequiredTicket	Integer	Y	
ImageLink	String	Y	
ImageLinkSmall	String	Y	
Type	String	Y	
GameCode	String	Y	
PrizePool	Decimal	Y	
NoOfSpins	Integer	Y	
TypeDescription	String	Y	
TypeName	String	Y	
PoolContribution	Decimal	Y	
Prize	Array	Y	
属性	类型	必填	说明
PrizeOCode	String	Y	
Amount	Decimal	Y	
Qty	Integer	Y	
Position	Integer	Y	
Ticket	Integer	Y	
Points	Long	Y	

5.4. 获取锦标赛等级

请求

POST {api-url}/tournament-rank

Content-Type: application/json

```
{
  "AppID": "AppID",
  "TournamentID": "xeahz7p7oag51",
  "PageIndex": 0,
  "Size": 20,
  "Timestamp": "1625640447662",
  "Hash": "4492ab519ea0038e2651ea40abc236a5"
}
```

属性	类型	必填	说明
AppID	String	Y	
TournamentID	String	Y	
PageIndex	Integer	Y	
Size	Integer	Y	
Hash	String	Y	
Timestamp	Long	Y	UNIX 时间戳

响应

```

{
  "Error": "0",
  "Description": "OK",
  "Data": [
    {
      "Username": "0000000A012",
      "DisplayName": "Tester",
      "CurrencyCode": "MYR",
      "Chance": 40770,
      "Order": 1,
      "PrizeName": "First Prize",
      "WinningAmount": 10000,
      "IsWinner": true,
      "Description": null,
      "JoinTime": "2021-06-18T16:23:23.242743",
      "Tickets": 0
    },
    .....
  ]
}

```

属性	类型	必填	说明
Error	String	Y	成功 = “0” 失败 = “1”
Description	String	Y	
Data	Array	Y	
属性	类型	必填	说明
Username	String	Y	
DisplayName	String	Y	
CurrencyCode	String	Y	
Chance	Integer	Y	
Order	Integer	Y	
PrizeName	String	Y	
WinningAmount	Decimal	Y	
IsWinner	Boolean	Y	
Description	String	Y	
JoinTime	Datetime	Y	
Tickets	Integer	Y	

6. 运营商集成 APIs

6.1. 验证令牌

请求

POST {operator-integration-api-url}/authenticate-token
Content-Type: application/x-www-form-urlencoded

请求内容:

appid=seamless&hash=1f4f020546a137427f4de3773dc04fc3&ip=128.199.64.190×tamp=1552970614889&token=xm6rhmcjweuq

属性	类型	必填	说明
token	String	Y	运营商玩家会话的唯一标识符
ip	String	Y	玩家 IP
timestamp	Long	Y	UNIX 时间戳
appid	String	Y	
hash	String	Y	

响应: Content-Type: application/json

```
{
  "Username": "DEMO", (4 to 32 alphanumeric characters and case-insensitive)
  "Balance": 100.01,
  "Message": "Success",
  "Status": 0
}
```

6.2. 验证用户名/密码 (可选)

请求

POST {operator-integration-api-url}/authenticate
Content-Type: application/x-www-form-urlencoded

请求内容:

appid=seamless&hash=30795ed16135a0eb7711229c95670924&ip=128.199.64.190&password=abcdef×tamp=1552976364273&username=DEMO

属性	类型	必填	说明
username	String	Y	玩家用户名 (4 to 32 alphanumeric characters and case-insensitive)
password	String	Y	玩家密码
ip	String	Y	玩家 IP
timestamp	Long	Y	UNIX 时间戳
appid	String	Y	

hash	String	Y
------	--------	---

响应: Content-Type: application/json

```
{"Token":"xm6rhmjciweuq","Balance":100.05,"Message":"Success","Status":0}
```

6.3. 余额

请求

POST {operator-integration-api-url}/balance

Content-Type: application/x-www-form-urlencoded

请求内容:

```
appid=seamless&hash=45a8710d5db0aeabf4f3e3d881f732f8&timestamp=1554370334626&username=DEMO
```

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
timestamp	Long		Y	
username	String		Y	

响应: Content-Type: application/json

```
{"Balance":1186.11,"Message":"Success","Status":0}
```

6.4. 下注

请求

POST {operator-integration-api-url}/bet

Content-Type: application/x-www-form-urlencoded

请求内容:

```
amount=0.50&appid=seamless&gamecode=zygj7oqga9nck&hash=41399ea6eeedd294fa0dfc0b911895e7&id=B_Tb5aa0018c11e&roundid=qc9cfdk7u5wca&timestamp=1552970894238&username=DEMO
```

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
id	String		Y	BetID -会员将会是独特性
amount	Decimal	Positive numeric	Y	
username	String		Y	
timestamp	Long		Y	
gamecode	String		Y	
roundid	String		Y	会员将会是独特性

响应: Content-Type: application/json

```
{"Balance":1186.11,"Message":"Success","Status":0}
```

如果下注超时，将发送取消。

6.5. 下注结算

请求

POST {operator-integration-api-url}/settle-bet
Content-Type: application/x-www-form-urlencoded

请求内容:

```
amount=1.00&appid=seamless&gamecode=zygj7oqga9nck&hash=ea28cd5801d5914e027d31275f5670e4&id=S_Tb5aa0018c11f&roundid=qc9cfdk7u5wca&timestamp=1552970900488&username=DEMO&description=slot&type=main
```

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
id	String		Y	偿付/结算 ID -会员将会是独特性
amount	Decimal	Positive numeric	Y	
username	String		Y	
timestamp	Long		Y	
gamecode	String		Y	
roundid	String		Y	
description	String		Y	
type	String		Y	

响应: Content-Type: application/json

```
{"Balance":1186.11,"Message":"Success","Status":0}
```

6.6.取消下注

请求

```
POST {operator-integration-api-url}/cancel-bet
Content-Type: application/x-www-form-urlencoded
```

请求内容:

```
appid=seamless&gamecode=zygj7oqga9nck&hash=ea28cd5801d5914e027d31275f5670e4&id=C_Tb5aa0018c11f&betid=B_Tb5aa0018c11e&roundid=qc9cfdk7u5wca&timestamp=1552970900488&username=DEMO
```

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
id	String		Y	取消下注 ID -会员将会是独特性
username	String		Y	
timestamp	Long		Y	
gamecode	String		Y	
roundid	String		Y	会员将会是独特性
betid	String		Y	

响应: Content-Type: application/json

```
{"Balance":1186.11,"Message":"Success","Status":0}
```

6.7.赢取奖金

请求

```
POST {operator-integration-api-url}/bonus-win
Content-Type: application/x-www-form-urlencoded
```

请求内容:

```
amount=1.00&appid=seamless&gamecode=zygj7oqga9nck&hash=ea28cd5801d5914e027d31275f5670e4&id=Tb5aa0018c11f&roundid=qc9cfdk7u5wca&timestamp=1552970900488&username=DEMO&description=bonus&type=bonus
```

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	

id	String	Y	会员将会是独特性
amount	Decimal	Positive numeric	Y
username	String	Y	
timestamp	Long	Y	
gamecode	String	Y	
roundid	String	Y	会员将会是独特性
description	String	Y	
type	String	Y	

响应: Content-Type: application/json

```
{"Balance":1186.11,"Message":"Success","Status":0}
```

6.8. 赢取头奖

请求

POST {operator-integration-api-url}/jackpot-win
Content-Type: application/x-www-form-urlencoded

请求内容:

```
amount=1.00&appid=seamless&gamecode=zygj7oqga9nck&hash=ea28cd5801d5914e027d31275f5670e4&id=Tb5aa0018c11f&roundid=qc9cfdk7u5wca&timestamp=1552970900488&username=DEMO&description=jackpot&type=jackpot
```

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
id	String		Y	会员将会是独特性
amount	Decimal	Positive numeric	Y	
username	String		Y	
timestamp	Long		Y	
gamecode	String		Y	默认: qc8y6dypyeboy
roundid	String		Y	会员将会是独特性
description	String		Y	
type	String		Y	

响应: Content-Type: application/json

```
{"Balance":1186.11,"Message":"Success","Status":0}
```


6.9. 事务

请求

POST {operator-integration-api-url}/transaction
Content-Type: application/x-www-form-urlencoded

请求内容:

amount=6.00&appid=seamless&description=Main&endbalance=318.00&gamecode=kk8nqm3cfwtng&hash=00b319f583ce6b2a26c2df0fe75f0d47&id=xm6qbj99cmfdq&result=4.00&roundid=xm6qbj99cmfdq&startbalance=320.00×tamp=1555474256669&type=Main&username=DEMO

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
id	String		Y	会员将会是独特性
amount	Decimal	Positive numeric	Y	
result	Decimal	Positive numeric	Y	
username	String		Y	
timestamp	Long		Y	
gamecode	String		Y	
roundid	String		Y	会员将会是独特性
description	String		Y	
type	String		Y	
startbalance	Decimal	Positive numeric	Y	
endbalance	Decimal	Positive numeric	Y	

响应: Content-Type: application/json

{"Balance":1186.11,"Message":"Success","Status":0}

6.10. 提现

请求

POST {operator-integration-api-url}/withdraw
Content-Type: application/x-www-form-urlencoded

请求内容:

amount=1.00&appid=seamless&hash=ea28cd5801d5914e027d31275f5670e4&id=Tb5aa0018c11f×tamp=1552970900488&username=DEMO

属性	类型	值	必填	说明
appid	String		Y	

hash	String		Y	
id	String		Y	会员将会是独特性
amount	Decimal	Positive numeric	Y	
username	String		Y	
timestamp	Long		Y	

响应: Content-Type: application/json

```
{"Balance":1186.11,"Message":"Success","Status":0}
```

6.11. 充值

请求

POST {operator-integration-api-url}/deposit
Content-Type: application/x-www-form-urlencoded

请求内容:

```
amount=1.00&appid=seamless&hash=ea28cd5801d5914e027d31275f5670e4&id=Tb5aa0018c11f&timestamp=1552970900488&username=DEMO
```

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
id	String		Y	会员将会是独特性
amount	Decimal	Positive numeric	Y	
username	String		Y	
timestamp	Long		Y	

响应: Content-Type: application/json {"Balance":1186.11,"Message":"Success","Status":0}

7. 运营商集成锦标赛 API

7.1. 加入锦标赛

当玩家加入锦标赛时，会调用 Join Tournament。运营商预计会扣除玩家余额并记录交易

请求

POST {operator-integration-api-url}/join-tournament
Content-Type: application/x-www-form-urlencoded

请求内容:

amount=10.00&appid=seamless&extendedinfo=%7B%22PoolContribution%22%3A2.00%7D&hash=c58166e6524e28cbdb47037133a8f83a&id=jxeahin3up4q51×tamp=1625644814173&tournamentid=xeahin3up4q51&username=DEMO

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
id	String		Y	会员将会是独特性
tournamentid	String		Y	
amount	Decimal	Positive numeric	Y	
username	String		Y	
timestamp	Long		Y	
extendedinfo	String		Y	<p>-不要在 Hash 中使用此属性</p> <p>- 是 JSON 字符串。如果需要消费，客户端可以将此值反序列化为对象。 例如：JSON.parse(value)</p>

响应: Content-Type: application/json {"Balance":1186.11,"Message":"Success","Status":0}

7.2.取消加入锦标赛

运营商应当退还玩家余额（如果玩家参加锦标赛时扣除余额），并取消其交易

请求

POST {operator-integration-api-url}/cancel-join-tournament
Content-Type: application/x-www-form-urlencoded

请求内容:

appid=seamless&extendedinfo=%7B%7D&hash=9bbf0aacc55559c922343588229dd905&id=cxeahin3up4q51&jointournamentid=xeahin3up4q51×tamp=1625645436243&username=DEMO

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
id	String		Y	会员将会是独特性
jointournamentid	String		Y	
username	String		Y	
timestamp	Long		Y	
extendedinfo	String		Y	<p>-不要在 Hash 中使用此属性</p> <p>- 是 JSON 字符串。如果需要消费，客户端可以将此值反序列化为对象。 例如：JSON.parse(value)</p>

响应: Content-Type: application/json {"Balance":1186.11,"Message":"Success","Status":0}

7.3. 锦标赛 Win

当锦标赛结束并且玩家赢得锦标赛时，会调用 Win Tournament。运营商有望添加玩家余额并记录交易

请求

POST {operator-integration-api-url}/win-tournament
Content-Type: application/x-www-form-urlencoded

请求内容:

amount=5000.00&appid=seamless&extendedinfo=%7B%22PoolContribution%22%3A2.00%7D&hash=c9d19dcb9cea559c3d111c1779002534&id=wxeahin3up4q51×tamp=1625646194210&tournamentid=xeahin3up4q51&username=DEMO

属性	类型	值	必填	说明
appid	String		Y	
hash	String		Y	
id	String		Y	会员将会是独特性
tournamentid	String		Y	
username	String		Y	
amount	Decimal	Positive numeric	Y	
timestamp	Long		Y	
extendedinfo	String		Y	<p>-不要在 Hash 中使用此属性</p> <p>- 是 JSON 字符串。如果需要消费，客户端可以将此值反序列化为对象。 例如：JSON.parse(value)</p>

响应: Content-Type: application/json {"Balance":1186.11,"Message":"Success","Status":0}

8. 参考

8.1. 时间戳

时间戳用作请求的一部分，以验证完整性。如果请求比时间戳晚很多，运营商应该拒绝请求。

时间戳被定义为自 1970 年 1 月 1 日午夜开始以来所经过的**毫秒数**（协调通用时间(UTC)）。

下面 C# 中的示例代码演示了如何获取当前时间戳：

```
public static readonly DateTime UnixEpoch = new DateTime(1970, 1, 1, 0, 0, 0, 0,
DateTimeKind.Local);
public static long GetCurrentTimestamp()
{
    return (long)DateTime.UtcNow.Subtract(UnixEpoch).TotalMilliseconds;
}
```

8.2. 散列

提供者利用 **MD5** 加密签名发出请求。例如，如果请求是：

```
{"key1": "value1", "key3": "value3", "key2": "value2", "key4": "" }
```

步骤 1： 构建查询字符串格式的原始签名字符串，并附加密钥。密钥为小写字母，并按字母顺序排序。没有值的密钥对将被忽略。**十进制数是两位十进制数字格式，例如 1.00**

```
String rawData = key1=value1&key2=value2&key3=value3secretKey
```

Example:

AppID: TXA2M

Timestamp: 1561025342810

Amount: **1000.0 (or 1000 or 1000.00)**

ID: xe6jrzgntw1pc

Username: DEMO

SecretKey: 123456

```
==> String rawData =
amount=1000.00&appid=TXA2M&id=xe6jrzgntw1pc&timestamp=1561025342810&username=DEMO1234
56
==> Hash = f38c4df2641a9c40129ae039a9ef8def
```

步骤 2： 使用密钥和 **MD5** 加密算法加密数据。

```
public static string GetMD5Signature(string rawData)
{
    var md5Content = string.Empty;
    using (MD5 md5Hash = MD5.Create())
    {
        md5Content = SignatureHelper.GetMd5Hash(md5Hash, rawData);
    }
    return md5Content;
}
```

}

```
private static string GetMd5Hash(MD5 md5Hash, string rawData)
{
    // Convert the input string to a byte array and compute the hash.
    byte[] data = md5Hash.ComputeHash(Encoding.UTF8.GetBytes(rawData));

    // Create a new StringBuilder to collect the bytes
    // and create a string.
    var sBuilder = new StringBuilder();

    // Loop through each byte of the hashed data
    // and format each one as a hexadecimal string.
    for (int i = 0; i < data.Length; i++)
    {
        sBuilder.Append(data[i].ToString("x2"));
    }

    // Return the hexadecimal string.
    return sBuilder.ToString();
}
```

注：签名可能包含一些特殊字符，应将其转换为用于查询字符串的转义表示形式。

8.3. 状态代码

状态	说明	适用的 API
0	成功	All
1	不允许 IP	All
2	无效的应用程序 ID	All
3	无效令牌	Authenticate-Token
4	无效参数	All
5	无效签名	All
6	无效时间戳	All
7	无效的用户名或密码	Authenticate
100	金额不足	Bet, Withdraw

201	正在处理事务	All
999	服务器正在维护中	All
1000	其他	All

9. 常见问题

1. 如何将“下注”与“结算”相关联？

结算可用于多次下注。下注和结算通过回合 ID 相关联。

2. 何时使用“赢取奖金” API？

奖金可以被认为是不需下注即可结算。第三方应在收到事件时为玩家添加余额。

3. 何时使用“赢取头奖” API？

头奖可以被认为是不需下注即可结算。第三方应在收到事件时为玩家添加余额。

4. “结算”后能收到“取消”吗？

可以。存在在结算后发出取消的情况。在这种情况下，应将余额添加给玩家。第三方应调整事务。

5. 何时使用“提现” API？

该 API 用于玩家转账以玩钓鱼游戏。

6. 何时使用“充值” API？

该 API 用于返还资金。

7. 何时使用“事务” API？

该 API 用于发送下注详细分组事务（钓鱼），其中在下注时不发送下注详细信息。API 对玩家余额没有影响。

8. 何时使用“正在处理事务”状态代码？

当 Provider 向第三方发送第一个请求，但处理的时间超过了超时时间时，Provider 将再次发送。

a. 正在处理第一个请求 -> 应返回第二个请求和状态 **201**

b. 第一个请求完成 -> 应返回第二个请求和正确的状态（使调用幂等）。