

PG SOFT

对接文档

转账模式 v2.2.12



目录

更新记录	4
1. 概述	8
2. 系统要求	9
2.1 游戏尺寸要求	9
2.2 硬件	9
2.3 浏览器	9
2.4 不支持的浏览器	10
3. 游戏启动	11
3.1 游戏登录方式	11
3.1.1 URL Scheme 方式	12
3.1.2 Web 大厅方式	17
3.2 游戏登录限制	19
4. 启动游戏 JS 库	20
4.1 避免浏览器弹出窗口受阻挡	20
4.2 实现方式	21
5. 对接 API	23
5.1 运营商提供对接 API	23
5.1.1 API 格式	23
5.1.2 令牌验证	27
5.2 PG SOFT 提供对接 API	30
5.2.1 API 格式	30
5.2.2 查询玩家的钱包余额	31
5.2.3 充值与转出流程	39
5.2.4 钱包操作限制及处理	44
5.2.5 转出所有余额（任选项）	45
5.2.6 获取单个交易记录（任选项）	47
5.2.7 投注详情页面（任选项）	49
5.2.8 获取多个玩家钱包余额（任选项）	52

5.2.9 其他（任选项）	54
5.3 PG SOFT 游戏对接运营商 iOS 应用程序.....	55
6. 解决待处理投注.....	56
7. 重置游戏状态.....	58
8. 限制	59
8.1 IP 限制.....	59
8.2 货币限制.....	59
9. 附录	60
9.1 错误代码.....	60
9.2 货币代码.....	62
9.3 投注类型.....	68
9.4 交易类型.....	69
9.5 平台.....	70
9.6 语言.....	71

更新记录

版本	更新日期	描述
2.2.0	2019-11-09	修订文件初版
2.2.1	2020-03-09	5.1.1 令牌验证 <ul style="list-style-type: none"> • 添加了关于 operator_player_session 参数的描述 5.2.10 获取单个交易记录 <ul style="list-style-type: none"> • 修订 API 返回参数
2.2.2	2020-09-24	3.2 游戏登陆限制 <ul style="list-style-type: none"> • 添加了新内容 5.2.4 钱包操作限制及处理 <ul style="list-style-type: none"> • 添加了新内容
2.2.3	2020-10-02	5.2.3 充值与转出流程 <ul style="list-style-type: none"> • 添加了新错误代码 5.2.3.1 充值 <ul style="list-style-type: none"> • 添加了新错误代码 5.2.3.2 转出余额 <ul style="list-style-type: none"> • 添加了新错误代码 5.2.11 获取单个交易记录 <ul style="list-style-type: none"> • 添加了新错误代码
2.2.4	2021-03-17	在 API URL 添加了跟踪 Id, 并修订 API URL 例子 <ul style="list-style-type: none"> • 5.2.1 登录游戏 • 5.2.2 查询钱包余额 • 5.2.3.1 充值 • 5.2.3.2 转出余额 • 5.2.5 创建玩家账号 (任选项) • 5.2.6 踢出玩家 (任选项) • 5.2.7 冻结玩家 (任选项) • 5.2.8 恢复玩家账号 (任选项)

		<ul style="list-style-type: none"> 5.2.9 查看玩家状态（任选项） 5.2.10 转出所有余额（任选项） 5.2.11 获取单个交易记录（任选项） 5.2.3.12 投注详情页面（任选项）
2.2.5	2021-03-18	5.2.10 转出所有余额（任选项） <ul style="list-style-type: none"> 移除在 API 请求中的 amount 参数 修订 API 请求参数范例 修订 API 返回参数范例
2.2.6	2021-03-22	9.2 货币代码 <ul style="list-style-type: none"> 修订货币名称
2.2.7	2021-03-28	添加新 API <ul style="list-style-type: none"> 5.2.13 获取用户钱包
2.2.8	2021-05-20	5.2.2 查询钱包余额 <ul style="list-style-type: none"> 添加了新 API 返回参数 修订 API 返回参数范例 8.2 货币限制 <ul style="list-style-type: none"> 修订货币限制列表 9.6 游戏 <ul style="list-style-type: none"> 修订游戏列表 5.2.10 转出所有余额 <ul style="list-style-type: none"> 修订 API 注解
2.2.9	2021-10-13	添加新内容 <ul style="list-style-type: none"> 5.1.1.3 哈希校验 修订限制的国家 <ul style="list-style-type: none"> 8.1 IP 限制 修订 API URL 和 API URL 示例中跟踪 ID <ul style="list-style-type: none"> 5.1.2 令牌验证 添加了新错误代码 <ul style="list-style-type: none"> 9.1 错误代码 修订浏览器支持版本

		<ul style="list-style-type: none"> 2.3 浏览器 修订请求 <ul style="list-style-type: none"> 3.1.1 URL Scheme 方式 移除列表 <ul style="list-style-type: none"> 9.6 游戏
2.2.10	2022-06-27	修订描述 <ul style="list-style-type: none"> 3.1.1 URL Scheme 方式 5.1.1.3.1 标头字段 添加新货币 <ul style="list-style-type: none"> 9.2 货币 添加新内容 <ul style="list-style-type: none"> 5.2.4 钱包操作限制及处理 2.4 不支持的浏览器 移除累积奖池内容 <ul style="list-style-type: none"> 5.2.14 其他（任选项） 修订描述以及 URL 范例 <ul style="list-style-type: none"> 5.1.1.3.1 标头字段 3.1.2 Web 大厅方式 移除 API 请求参数的 ct 参数 <ul style="list-style-type: none"> 3.1.1 URL Scheme 方式 添加新语言 <ul style="list-style-type: none"> 9.6 语言 修订 API URL <ul style="list-style-type: none"> 5.2.11 投注详情页面（任选项） 移除参数类型 <ul style="list-style-type: none"> 3.1.2 Web 大厅方式 移除玩家相关 API <ul style="list-style-type: none"> 5.2.6 创建玩家账号（任选项） 5.2.7 踢出玩家（任选项） 5.2.8 冻结玩家（任选项） 5.2.9 恢复玩家账号（任选项） 5.2.10 查看玩家状态（任选项）
2.2.11	2022-07-22	修订参数名称 <ul style="list-style-type: none"> 3.1.2 Web 大厅方式

		<ul style="list-style-type: none"> • 4.2 实现方式 修订内容及移除 PHP 货币 <ul style="list-style-type: none"> • 8.2 货币 移除 DEC 及 TNT 货币 <ul style="list-style-type: none"> • 9.2 货币代码 修订内容 <ul style="list-style-type: none"> • 3.1.1.1 现实查核 移除 rurl 参数 <ul style="list-style-type: none"> • 3.1.1 URL Scheme 方式 修订内容 <ul style="list-style-type: none"> • 5.1.1.3.1 标头字段 添加新内容 <ul style="list-style-type: none"> • 5.2.7 获取单个交易记录（任选项）
2.2.12	2023-05-31	添加新内容及新请求参数 <ul style="list-style-type: none"> • 3.1.1 URL Scheme 方式 • 3.1.2 Web 大厅方式 • 5.1.2 令牌验证 移除游戏登录方式 <ul style="list-style-type: none"> • 3.1 WebView 方式 • 3.1.3 WebView 方式（仅适用于 PG SOFT 应用程序和 PG SOFT 大厅） • 5.2.2 游戏登录 移除 ESP、HRK、STD、TRL 及 AUD 货币 <ul style="list-style-type: none"> • 9.2 货币

1. 概述

本文档详细介绍转账模式的对接方式。所谓转账模式，即指玩家需要从玩家钱包将金额转账 PG SOFT 游戏系统钱包才可进行游戏。这也表示，玩家需将盈利的金额从 PG SOFT 游戏系统钱包转账至中心钱包后才可进行提款。

2. 系统要求

2.1 游戏尺寸要求

PG SOFT 游戏以 H5 制作并以垂直方式呈现，在尺寸规格上有最低要求。有关 iframe 中的游戏尺寸，请参阅以下内容：

- 框架理想尺寸：1280 x 720
- 最小尺寸：540 x 960
- 最大尺寸：1080 x 2340

2.2 硬件

	Android	iOS	Windows
操作系统版本	5.0 或以上	10.0 或以上	7 或以上
内存	2GB	1GB	2GB
中央处理器	ARM Cortex-A9 或以上	iPhone 6 或以上	<ul style="list-style-type: none"> • 1 千兆赫 (GHz) 或更快 • 基于 32-bit (x86) 或 64-bit (x64) 处理器 • 至少双核
储存	2GB 和以上	2GB 和以上	20GB 和以上

2.3 浏览器

- Microsoft Edge：版本 79 或更高版本
- Firefox：版本 58 或更高版本
- Google Chrome：版本 59 或更高版本
- Safari：版本 11.1 或更高版本
- 其他：基于 Chromium 的浏览器需同等于 Google Chrome 59 或更高版本

2.4 不支持的浏览器

- Android : 火狐浏览器及 CM 浏览器
- iOS : CM 浏览器
- 桌面 : UC 浏览器、百度浏览器、夸克浏览器、CM 浏览器、IE 浏览器

3. 游戏启动

3.1 游戏登录方式

PG SOFT 支持两种登录授权方式：

- **URL Scheme** 方式适用于 iOS、Android 和浏览器，它使用令牌参数进行登录；
- **Web 大厅** 方式是一种响应性页面，它使用令牌参数进行登录。

3.1.1 URL Scheme 方式

URL scheme 方式是游戏启动的常规方法。玩家需要通过运营商的网站登录，并通过运营商的游戏大厅页面启动游戏。

注解:

运营商需提供令牌验证¹ API，并用于运营商玩家令牌的身份验证。

请求参数

API URL : {PgSoftPublicDomain}/{GameId}/index.html

HTTP method : GET

URL:

参数名	参数类型	必需项	描述
GameId	String	是	每个游戏的唯一标识符 ²

URL 参数:

参数名	参数类型	必需项	描述
btt	Integer	是	游戏启动模式 ³
ot	String	是	运营商独有的身份识别
ops	String	是	运营商系统生成的令牌 注: <ul style="list-style-type: none"> • 最多 200 字符 • 请将参数值设置为 GUID 格式或任何随机字符串 • 请使用 UriEncode 对值进行编码，以避免未知错误

¹ 请参考[令牌验证](#)的部分以获取更多信息

² 请从 PG SOFT 获取最新的列表

³ 请参考[投注类型](#)的部分以获取完整列表

l	String	否	游戏显示语言 ⁴ 默认：英文
te	Integer	否	运营商运行健康提醒的时间，以秒为单位（现实查核 ⁵ ）
ri	Integer	否	健康提醒的间隔时间，以秒为单位（现实查核 ⁵ ）
op	String	否	运营商的自定义参数，PG SOFT API 将在验证运营商玩家会话时包含参数值。 注： 请使用 UriEncode 对值进行编码，以避免未知错误
f	String	否	游戏退出 URL 默认：重定向到 PG SOFT 退出页面 注： <ul style="list-style-type: none"> 在试玩模式下，该数值将用于重定向到真实游戏的提示 分配数值到 <i>PGGameCloseUrl</i> 以关闭游戏窗口 请使用 UriEncode 对值进行编码，以避免未知错误
iwk	Integer	否	启动游戏时进行设备兼容性检查 ⁶ 0：普通模式（默认） 1：跳过兼容性检查 注： <ul style="list-style-type: none"> 跳过兼容性检查可能会导致 WebGL 错误或 WebGL 崩溃 仅适用于 IOS Web Kit 和 Safari
oc	Integer	否	启动游戏时检查屏幕方向 ⁷

⁴ 请参考[语言](#)的部分以获取完整列表

⁵ 请参考[现实查核](#)的部分以获取更多信息

⁶ 请参考截图

⁷ 请参考截图

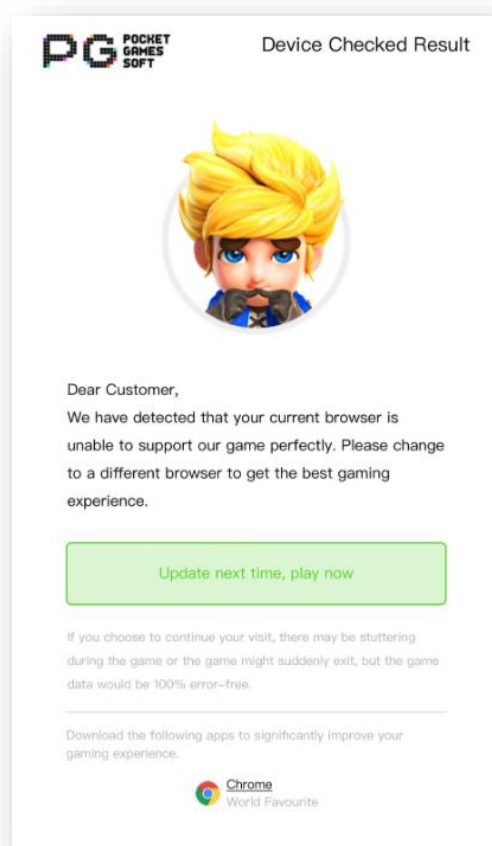
0: 跳过屏幕方向检查

1: 普通模式（默认）

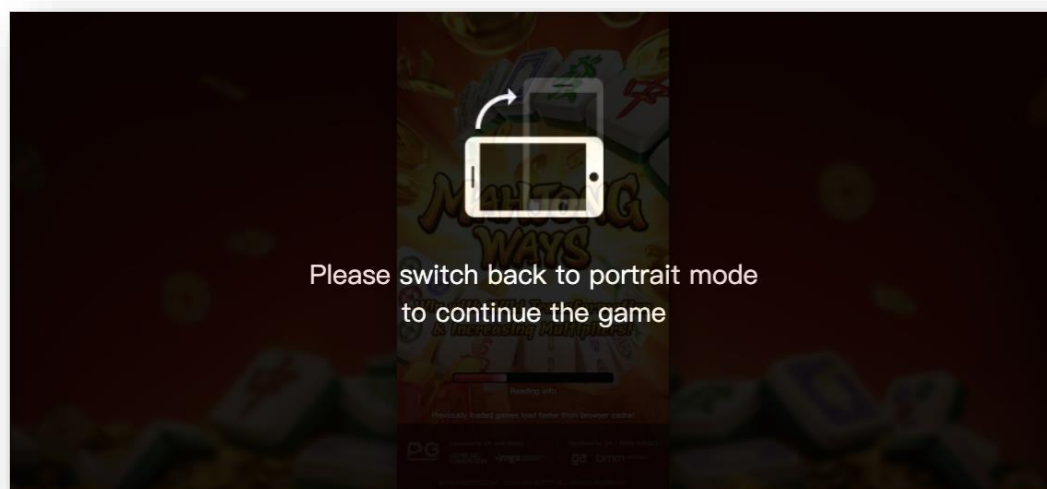
范例：

<https://m.pg-redirect.net/1/index.html?ot=abcd&ops=12345-abcd-1234-abcd-12345&btt=1>

截图



设备兼容性检查



屏幕方向检查

3.1.1.1 现实查核

于 URL scheme⁸的方式中, 运营商可将 “te” 和 “ri” 的参数包含在启动游戏的 URL 以进行现实查核。

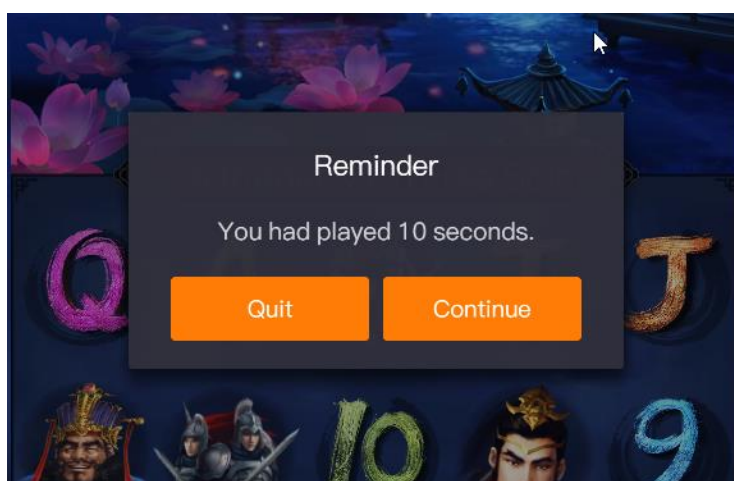
- te : 计算第一次现实查核所使用的时间 (以秒为单位)
- ri : 现实查核的间隔时间 (以秒为单位)

例子:

ri=300&te =290

第一个现实查核提醒将在 10 秒后显示 (300s-290s = 10s) , 后续提醒将每 5 分钟显示一次。

截图



健康提醒

⁸ 请参考 [URL Scheme 方式](#) 的部分以获取更多信息

3.1.2 Web 大厅方式

PG SOFT Web 大厅是个响应式的网页，玩家可以在这里玩到所有的 PG SOFT 游戏。它为玩家提供了全新的体验，呈现美观、具吸引力、丰富且有效率的 UI 设计。我们建议运营商将 PG SOFT Web 大厅整合到您的网站。

玩家需要通过运营商网站登录，并通过 PG SOFT Web 大厅启动 PG SOFT 游戏。

注解：

运营商需提供令牌验证⁹ API，用于运营商玩家令牌的身份验证。

请求参数

API URL : {PgSoftPublicDomain}/web-lobby/{panel_type}
 HTTP method : GET

URL:

参数名	参数类型	必需项	描述
panel_type	String	是	首次加载时 Web 大厅的显示面板 <i>[NO VALUE]</i> : 默认版面 (锦标赛) tournament/open : 锦标赛 – 开放报名的面板 tournament/process: 锦标赛 – 正在进行的面板 tournament/end : 锦标赛 – 结束的面板 games : 仅显示游戏

⁹ 请参考[令牌验证](#)的部分以获取更多信息

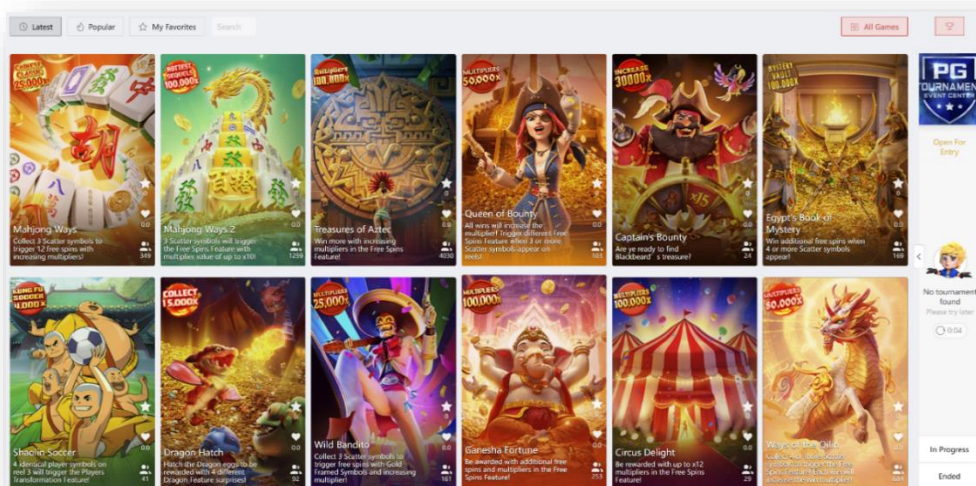
URL 参数:

参数名	参数类型	必需项	描述
ot	String	是	运营商独有的身份识别
ops	String	是	运营商系统生成的玩家令牌 注: <ul style="list-style-type: none"> 最多 200 字符 请将参数值设置为 GUID 格式或任何随机字符串 请使用 UriEncode 对值进行编码，以避免未知错误
l	String	否	Web 大厅的显示语言 ¹⁰ 默认值: 玩家设备的设置语言
width	Integer	否	Web 大厅的页面宽度 默认值: 响应式 数值范围: 数值≥930

范例:

<https://public.pg-redirect.net/web-lobby/tournament/open/?ot=abc123&ops=abc123-abc123&l=en>

截图



Web 大厅方式

¹⁰ 请参考[语言](#)的部分以获取完整列表

3.2 游戏登录限制

PG SOFT 游戏不支援一个账号多重登录或一个玩家同时打开多个游戏。以上操作将导致玩家被锁定，同时将被踢出游戏及报错：

S1302XXXXXXXX: 未能成功登录认证

S1309XXXXXXXX: 玩家账号被冻结

S1305XXXXXXXX: 服务器请求失败

若玩家在运行游戏时遇到以上错误，只需关闭其它游戏窗口并重启游戏，即可解锁他们的账号：

S1302XXXXXXXX: 请直接重新启动游戏

S1309XXXXXXXX: 玩家将与 10 分钟后解锁，请于 10 分钟后重新启动游戏

S1305XXXXXXXX: 请直接重新启动游戏

4. 启动游戏 JS 库

运营商需使用 PG SOFT JS 库启动游戏，以便为玩家提供最佳的游戏体验：

- 根据用户的显示器大小而自动调整弹出窗口。
- PG SOFT 游戏均设计为竖屏显示，弹出窗口的格式与产品设计一致。
- 实现最佳的游戏界面高度，并符合用户使用习惯，提升用户的感官享受。
- 向用户展示最高的游戏视觉质量，提供最佳的游戏体验。

4.1 避免浏览器弹出窗口受阻挡

弹出窗口被阻挡时所涉及的一般规则是：

- 从 JavaScript 调用，而不是直接从用户操作所调用。
- 弹出内容并不是源于同一个域名。

建议运营商遵循以下说明以避免浏览器弹出窗口受阻挡：

- 创建重定向页面并重定向到游戏启动 URL（URL scheme 方式¹¹）
- 当玩家点击游戏启动按钮或链接时，使用 PG SOFT JS 功能（PGSDK.launchGame）并启动重定向页面。

¹¹ 请参考 [URL Scheme 方式](#) 的部分以获取更多信息

4.2 实现方式

步骤 1

添加以下代码至 HTML <head> 标签：

```
<script defer src="https://public.pg-  
redirect.net/sdk/main.min.js"></script>
```

注解：

运营商也可以自行托管 JS 库：<https://public.pg-redirect.net/sdk/main.min.js>

步骤 2

运营商需要创建自行托管的重定向页面以绕过网络浏览器上的弹出窗口受阻挡。

重定向页面的示例代码：

```
<!DOCTYPE html>  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
<script>  
    window.location.replace("https://m.pg-  
redirect.net/1/index.html?l=en&btt=1&ot=12345abcde&ops=abcde-12345-abcde-  
12345");  
</script>  
</head>  
</html>
```

步骤 3

使用以下 JS script 以启动 PG SOFT 游戏。运营商需要创建自行托管的重定向页面以绕过网络浏览器上的弹出窗口受阻挡。

```
PGSDK.launchGame('{operator_redirection_URL}');
```

参数:

参数名	参数类型	必需项	描述
operator_redirection_URL	String	是	运营商自托管页面, 它将重定向到游戏启动 URL (URL scheme 方式 ¹²)

范例:

```
PGSDK.launchGame('https://[OperatorDomain]/redirect.html');
```

¹² 请参考 [URL Scheme 方式](#) 的部分以获取更多信息

5. 对接 API

5.1 运营商提供对接 API

5.1.1 API 格式

5.1.1.1 请求参数

该 API 使用 HTTP 表单方式以及 RESTful 终点架构。PG SOFT 将根据以下内容格式发送请求：

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

5.1.1.2 返回结果

API 返回 JSON 格式的返回结果。对于成功和失败的 API 请求，运营商需返回 HTTP 状态码为 200 的 API 响应结果。

API 对于全部 API 方式都使用全球 JSON 格式的返回结果，其内容类型如下：

`Content-Type: application/json`

参数：

参数名	参数类型	描述
data ¹³	JSON Object	API 方式的返回结果信息
error	JSON Object	发生错误或异常时的错误信息

范例：

成功返回

```
{
  "data": {
    [API 返回信息。JSON 格式将根据各个 API 而有所不同]
  },
  "error": null,
}
```

失败/错误返回

```
{
  "data": null,
  "error": {
    "code": "[错误代码]",
    "message": "[错误信息]"
  }
}
```

¹³ 该部分将在错误返回中显示空值

5.1.1.3 哈希校验（任选项）

我们强烈建议运营商使用哈希校验以保护每个请求。若运营商选择使用哈希校验，PG SOFT 系统将对每个请求头字段的哈希信息进行验证。

5.1.1.3.1 标头字段

下表描述了下述示例中的各种请求头：

构件	描述
Host	API 域名 例子： <code>apiexample.pgsoft.com</code>
x-date	以 YYYYMMDD 为格式的当前日期（UTC） 例子： <code>20190902</code>
x-content-sha256	请求主体内容字符串的 SHA256 哈希值 例子： <code>1700116101f424b9f6fc695b4dbaf2b7b0ee763ba1b3b53298e3069143ed46f1</code>
Authorization	多个组成部分的组合，用于请求认证 例子： <code>PWS-HMAC-SHA256 Credential=20190902/OPERATOROKENEXAMPLE/pws/v1,SignedHeaders=host;x-content-sha256;x-date,Signature=d78220cf06ae85f9d1db11dad9c3fd926799619eab3d28574aadb8cf328cd7aa</code>

范例

```
Host:
apiexample.pgsoft.com
x-date: 20190902
x-content-sha256:
1700116101f424b9f6fc695b4dbaf2b7b0ee763ba1b3b53298e3069143ed46f1
Authorization: PWS-HMAC-SHA256
Credential=20190902/OPERATORTOKENEXAMPLE/pws/v1,SignedHeaders=host;x-
content-sha256;x- date,Signature=
d78220cf06ae85f9d1db11dad9c3fd926799619eab3d28574aadb8cf328cd7aa
```

请求头: x-content-sha256

计算请求主体内容字符串的 SHA256 哈希值:

范例

POST 主体:

```
operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&count=5000&bet_type=1&
row_version=1346592723000
```

x-content-sha256:

```
1700116101f424b9f6fc695b4dbaf2b7b0ee763ba1b3b53298e3069143ed46f1
```

请求头: 验证

下表描述了下述示例中 Authorization 标头值的各个组成部分:

构件	描述
Credential	<p>您的访问密钥 ID 和范围信息, 包括用于计算签名的日期、运营商令牌和服务。</p> <p>该字符串具有以下形式:</p> <pre>{x-date}/{operator_token}/pws/v1</pre> <p>例子:</p> <pre>20190902/OPERATORTOKENEXAMPLE/pws/v1</pre>
SignedHeaders	<p>以分号分隔的请求头列表, 用于计算签名。该列表仅包含标头名称, 并且标头名称必须为小写。</p>

	固定值: <code>host;x-content-sha256;x-date</code>
Signature	以 64 个小写的十六进制字符表示的 256-bit 签名。 例子: <code>d78220cf06ae85f9d1db11dad9c3fd926799619eab3d28574aad db8cf328cd7aa</code>

要计算签名，您首先需要包含 `{host}` `{x-content-sha256}` `{x-date}` 的签名字符串。
 接着，您将可以使用签名密钥计算出签名字符串的 HMAC-SHA256 哈希值：

```
hmac-sha256({salt}, {host}{x-content-sha256}{x-date})
```

计算主体内容字符串的 HMAC-SHA256 哈希值：

例子

```
Salt           : SALTEXAMPLE
Host           : apiexample.pgsoft.com
x-date        : 20190902
x-content-sha256 :
1700116101f424b9f6fc695b4dbaf2b7b0ee763ba1b3b53298e3069143ed46f1
```

签名:

```
d78220cf06ae85f9d1db11dad9c3fd926799619eab3d28574aaddb8cf328cd7aa
```

5.1.2 令牌验证

PG SOFT 游戏系统将调用此运营商 API 以检查玩家登录游戏时所生成的玩家令牌。

注解:

- URL Scheme 和 Web 大厅的授权模式需要使用此 API¹⁴
- 在此模式下系统将自动创建新玩家

请求参数

API URL : {OperatorAPIDomain}/VerifySession
HTTP method : POST

URL 参数:

参数名	参数类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: 请把参数值设置为 GUID 格式

主体参数:

参数名	参数类型	必需项	描述
operator_token	String	是	运营商独有的身份识别
secret_key	String	是	PG SOFT 与运营商之间共享密码
operator_player_session	String	是	游戏启动时, 运营商系统将生成令牌 (URL Scheme 方式 ¹⁵)
bet_type	Integer	是	游戏启动模式 ¹⁶
ip	String	否	玩家 IP 地址
custom_parameter	String	否	URL Scheme 方式 ¹⁷ 中的 <i>operator_param</i> 值

¹⁴ 请参考 [URL Scheme 方式](#) 和 [Web 大厅](#) 的部分以获取登录流程信息

¹⁵ 请参考 [URL Scheme 方式](#) 的部分以获取更多信息

¹⁶ 请参考 [投注类型](#) 的部分以获取完整列表

¹⁷ 请参考 [URL Scheme 方式](#) 的部分以获取更多信息

game_id	Integer	否	游戏的独有代码
---------	---------	---	---------

范例:

URL:

https://operator.api/VerifySession?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&operator_player_session=a2b3c4d5e6f7g8&ip=1.1.1.1&custom_parameter=operatorparam123&game_id=1&bet_type=1

验证方式

运营商需执行以下参数验证:

参数名	描述	预期结果
operator_token	确保参数值与 PG SOFT 所提供的完全相同	不匹配时返回错误
secret_key		
operator_player_session	确保运营商玩家会话与 URL Scheme 方式 ¹⁸ 中所生成的令牌相同	

返回结果

主体参数:

JSON 对象

参数名	参数类型	必需项	描述
player_name	String	是	玩家帐号 <ul style="list-style-type: none"> 玩家名称不区分大小写 仅允许使用字母, 数字以及 "@"、"-","_" 符号 注: 上限 50 字符
nickname	String	否	玩家昵称 注: 上限 50 字符
currency	String	是	玩家选择的币种

¹⁸ 请参考 [URL Scheme 方式](#) 的部分以获取更多信息

范例:

```
{
  "data": {
    "player_name": "player1",
    "nickname": "nickname",
    "currency": "USD"
  },
  "error": null
}
```

错误返回

代码	信息
1034	无效请求
1200	内部服务器错误

5.2 PG SOFT 提供对接 API

5.2.1 API 格式

5.2.1.1 请求参数

该 API 使用 HTTP 表单方式以及 RESTful 终点架构。PG SOFT 将根据以下内容格式发送请求：

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

5.2.1.2 返回结果

API 返回 JSON 格式的返回结果。对于成功和失败的 API 请求，运营商需返回 HTTP 状态码为 200 的 API 响应结果。

API 对于全部 API 方式都使用全球 JSON 格式的返回结果，其内容类型如下：

`Content-Type: application/json`

参数：

参数名	参数类型	描述
<code>data</code> ¹⁹	JSON Object	API 方式的返回结果信息
<code>error</code> ²⁰	JSON Object	发生错误或异常时的错误信息

范例：

成功返回

```
{
  "data": {
    [API 返回信息。JSON 格式将根据各个 API 而有所不同]
  },
  "error": null,
}
```

失败/错误返回

```
{
  "data": null,
  "error": {
    "code": "[ 错误代码 ]",
    "message": "[ 错误信息 ]"
  }
}
```

¹⁹该部分将在错误返回中显示空值

²⁰ 该部分将在成功返回 API 中显示空值。请参考[错误代码](#)的部分以获取错误返回的格式

5.2.2 查询玩家的钱包余额

查询特定玩家的钱包余额。

请求参数

API URL : {PgSoftAPIDomain}/Cash/v3/GetPlayerWallet

HTTP method : POST

URL 参数:

参数名	参数类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: 请把参数值设置为 GUID 格式

主体参数:

参数名	参数类型	必需项	描述
operator_token	String	是	运营商独有的身份识别
secret_key	String	是	PG SOFT 与运营商之间共享密码
player_name	String	是	玩家独有的身份识别

范例:

URL:

https://api.pg-bo.me/external/Cash/v3/GetPlayerWallet?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123

返回结果

主体参数：

JSON 对象

参数名	参数类型	描述
currencyCode	String	玩家选择的币种
totalBalance	Decimal	所有玩家钱包余额的总和： <ul style="list-style-type: none"> • 现金余额 • 红利余额 • 免费游戏余额
cashBalance	Decimal	玩家的现金钱包余额 运营商可参考该值作为玩家余额
totalBonusBalance	Decimal	玩家的红利余额 这将显示玩家的所有可用红利余额，它无法被转出。
freeGameBalance	Decimal	玩家的免费游戏钱包余额 这将显示玩家的所有可用的免费游戏，它无法被转出。
bonuses	JSON Array	玩家的红利详情
freeGames	JSON Array	玩家的免费游戏详情
cashWallet	JSON Object	玩家的现金钱包详情
key	String	现金钱包的唯一标识符
cashId	Integer	玩家现金钱包的唯一标识符
cashBalance	Decimal	玩家的现金余额
bonusWallet	JSON Object	玩家的红利钱包详情
totalBonusBalance	Decimal	玩家的红利钱包余额总和
freeGameWallet	JSON Object	玩家的免费游戏钱包详情
freeGameBalance	Decimal	玩家的免费游戏钱包余额总和

JSON 数组（红利游戏）

参数名	参数类型	描述
key	String	红利钱包的唯一标识符

bonusId	Integer	红利钱包的唯一标识符 只有在红利游戏分配给玩家后才会出现
bonusName	String	红利游戏名称 只有在红利游戏分配给玩家后才会出现
transactionId	String	交易的唯一标识符
bonusParentType	String	母红利类型： 红利：正常创建的红利游戏 免费游戏：从免费游戏转换的红利游戏
gameIds	Integer Array	游戏的独有代码
gameIdLock	Integer	未完成的游戏 ID
balanceAmount	Decimal	玩家的红利余额
bonusRatioAmount	Decimal	红利游戏的流水要求 指红利游戏里的所赢金额在进入红利钱包之后，需要完成的总金额
minimumConversionAmount	Decimal	红利游戏的最低现金转换金额
maximumConversionAmount	Decimal	红利游戏的最高现金转换金额
status	Integer	红利钱包的状态： 0：无效（已取消） 1：活跃 2：已过期 3：已转换 4：已完成 5：全新 6：已取消用户 7：已锁定 8：待处理
createdDate	Long	红利游戏的创建日期

		(Unix 时间戳, 以毫秒为单位)
expiredDate	Long	红利游戏的截止日期 (Unix 时间戳, 以毫秒为单位)
createdBy	String	创建红利游戏的 API 或 BackOffice 用户
updatedBy	String	更新红利游戏的 API 或 BackOffice 用户
isSuppressDiscard	Boolean	允许玩家取消优惠: True: 不允许玩家取消优惠 False: 允许玩家取消优惠

JSON 数组 (免费游戏)

参数名	参数类型	描述
key	String	免费游戏钱包的唯一标识符
freeGameId	Integer	免费游戏的唯一标识符 只有在免费游戏分配给玩家后才会出现
freeGameName	String	免费游戏名称 只有在免费游戏分配给玩家后才会出现
transactionId	String	交易的唯一标识符
gameIds	Integer Array	游戏的独有代码
gameIdLock	Integer	未完成的游戏 ID
gameCount	Integer	剩余的免费游戏总数
totalGame	Integer	免费游戏总数
balanceAmount	Decimal	玩家的免费游戏余额
minimumConversionAmount	Decimal	免费游戏的最低现金转换金额
maximumConversionAmount	Decimal	免费游戏的最高现金转换金额
multiplier	Integer	免费游戏的投注倍数
coinSize	Decimal	免费游戏的筹码大小
createdDate	Long	免费游戏的创建时间 (Unix 时间戳, 以毫秒为单位)

expiredDate	Long	免费游戏的截止时间 (Unix 时间戳, 以毫秒为单位)
status	Integer	免费游戏钱包的状态: 0: 无效 (已取消) 1: 活跃 2: 已过期 3: 已转换 4: 已完成 5: 全新 6: 已取消用户 7: 已封锁 8: 待处理
conversionType	Char	免费游戏完成后的转换类型: C: 现金 B: 红利
createdBy	String	在 API 或 BackOffice 创建免费游戏的用户
updatedBy	String	在 API 或 BackOffice 更新免费游戏的用户
isSuppressDiscard	Boolean	允许玩家取消优惠: True: 不允许玩家取消优惠 False: 允许玩家取消优惠

范例:

```
{
  "data": {
    "currencyCode": "CNY",
    "totalBalance": 50732.39,
    "cashBalance": 48853.39,
    "totalBonusBalance": 1815.00,
    "freeGameBalance": 64.00,
    "bonuses": [
      {
        "key": "11019-97090_B",
        "bonusId": 11019,
        "bonusName": "PG Bonus ",
        "transactionId": "",
        "bonusParentType": "Bonus",
        "gameIds": [
          94,
          69,
          71
        ],
        "gameIdLock": 0,
        "balanceAmount": 1000.00,
        "bonusRatioAmount": 1000.00,
        "minimumConversionAmount": 0.00,
        "maximumConversionAmount": 0.00,
        "status": 5,
        "createdDate": 1615952183000,
        "expiredDate": 1640966399000,
        "createdBy": "operator1",
        "updatedBy": "operator1",
        "isSuppressDiscard": true
      }
    ],
    "freeGames": [
      {
        "key": "575-16084_G",
        "freeGameId": 575,
        "freeGameName": "FREE GAME",
        "transactionId": "",
        "gameIds": [
          33
        ],
        "gameIdLock": 0,
        "gameCount": 0,
        "totalGame": 100,
        "balanceAmount": 0.00,
        "minimumConversionAmount": 0.00,
        "maximumConversionAmount": 0.00,
        "multiplier": 6,
        "coinSize": 0.1,
        "createdDate": 1544582329000,
        "expiredDate": 1544630399000,
        "status": 3,
        "conversionType": "C",
        "createdBy": "operator2",

```

```

        "updatedBy": "operator2",
        "isSuppressDiscard": false
      }
    ],
    "cashWallet": {
      "key": "0_C",
      "cashId": 0,
      "cashBalance": 48853.39
    },
    "bonusWallet": {
      "totalBonusBalance": 1815.00,
      "bonuses": [
        {
          "key": "11019-97090_B",
          "bonusId": 11019,
          "bonusName": "PG Bonus ",
          "transactionId": "",
          "bonusParentType": "Bonus",
          "gameIds": [
            94,
            69,
            71
          ],
          "gameIdLock": 0,
          "balanceAmount": 1000.00,
          "bonusRatioAmount": 1000.00,
          "minimumConversionAmount": 0.00,
          "maximumConversionAmount": 0.00,
          "status": 5,
          "createdDate": 1615952183000,
          "expiredDate": 1640966399000,
          "createdBy": "operator3",
          "updatedBy": "operator3",
          "isSuppressDiscard": true
        }
      ]
    },
    "freeGameWallet": {
      "freeGameBalance": 64.00,
      "freeGames": [
        {
          "key": "575-16084_G",
          "freeGameId": 575,
          "freeGameName": "FREE GAME",
          "transactionId": "",
          "gameIds": [
            33
          ],
          "gameIdLock": 0,
          "gameCount": 0,
          "totalGame": 100,
          "balanceAmount": 0.00,
          "minimumConversionAmount": 0.00,
          "maximumConversionAmount": 0.00,
          "multiplier": 6,
          "coinSize": 0.1,

```

```

    "createdDate": 1544582329000,
    "expiredDate": 1544630399000,
    "status": 3,
    "conversionType": "B",
    "createdBy": "operator4",
    "updatedBy": "operator4",
    "isSuppressDiscard": false
  }
]
},
"error": null
}

```

错误返回

代码	信息
3001	不能空值
3005	玩家钱包不存在

5.2.3 充值与转出流程

PG SOFT 系统正在为所有 API 操作实现**幂等操作**。根据以下情况，要求运营商通过使用相同的 transfer_reference **重新发送**请求：

- 从 PG SOFT 系统中收到错误的返回（错误的返回格式）。
- 收到除了 3001、3005、3100 的错误返回代码。
- 没有收到 PG SOFT 系统的返回。

5.2.3.1 充值

充值现金至特定玩家的现金钱包。

注解:

- Transfer_reference 是每笔交易的唯一识别码。运营商需要为每笔传送请求发送新的 transfer_reference。
- 对于重复请求，PG SOFT 不对玩家钱包余额进行任何操作，但返回转账成功的响应。

请求参数

API URL : {PgSoftAPIDomain}/Cash/v3/TransferIn

HTTP method : POST

URL 参数:

参数名	参数类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: 请把参数值设置为 GUID 格式

主体参数:

参数名	参数类型	必需项	描述
operator_token	String	是	运营商独有的身份识别
secret_key	String	是	PG SOFT 与运营商之间共享密码
player_name	String	是	玩家账号
amount	Decimal	是	转账金额 (小数点后两位) 注: 最低 0.01
transfer_reference	String	是	交易凭证, 运营商平台生成并确保每次交易是独有的。 注: 上限 50 字符
currency	String	是	玩家选择的币种

范例:

URL:

https://api.pg-bo.me/external/Cash/v3/TransferIn?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123&amount=0.30&transfer_reference=ref1530879795000¤cy=EUR

返回结果

主体参数:

JSON 对象

参数名	参数类型	描述
transactionId	String	特殊交易识别码
balanceAmountBefore	Decimal	交易前的玩家余额
balanceAmount	Decimal	交易后的玩家余额
amount	Decimal	交易金额

范例:

```
{
  "data": {
    "transactionId": 35509540,
    "balanceAmountBefore": 1000.00,
    "balanceAmount": 1100.00,
    "amount": 100.00
  },
  "error": null
}
```

错误返回

代码	信息
3001	不能空值
3005	玩家钱包不存在
3100	转账失败
3101	转账请求中, 请重试查看最新状态

5.2.3.2 转出余额

将特定玩家的现金钱包的余额转出。

注解:

- Transfer_reference 是每笔交易的唯一识别码。运营商需要为每笔传送请求发送新的 transfer_reference。
- 对于重复请求，PG SOFT 不对玩家钱包余额进行任何操作，但返回转账成功的响应。

请求参数

API URL : {PgSoftAPIDomain}/Cash/v3/TransferOut

HTTP method : POST

URL 参数:

参数名	参数类型	必需项	描述
trace_id	String	是	请求的唯一标识符 (GUID) 注: • 请把参数值设置为 GUID 格式

主体参数:

参数名	参数类型	必需项	描述
operator_token	String	是	运营商独有的身份识别
secret_key	String	是	PG SOFT 与运营商之间共享密码
player_name	String	是	玩家账号
amount	Decimal	是	转账金额 (小数点后两位) 注: 最低 0.01
transfer_reference	String	是	交易凭证, 运营商平台生成并确保每次交易是独有的。 注: 上限 50 字符
currency	String	是	玩家选择的币种

范例:

URL:

https://api.pg-bo.me/external/Cash/v3/TransferOut?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123&amount=0.30&transfer_reference=ref1530879795000¤cy=EUR

返回结果

主体参数:

JSON 对象

参数名	参数类型	描述
transactionId	String	特殊交易识别码
balanceAmountBefore	Decimal	交易前的玩家余额
balanceAmount	Decimal	交易后的玩家余额
amount	Decimal	交易金额

范例:

```
{
  "data": {
    "transactionId": 35509540,
    "balanceAmountBefore": 1000.00,
    "balanceAmount": 1100.00,
    "amount": 100.00
  },
  "error": null
}
```

错误返回

代码	信息
3001	不能空值
3005	玩家钱包不存在
3100	转账失败
3101	转账请求进行中, 请重试查看最新状态

5.2.4 钱包操作限制及处理

为了避免玩家余额报错，同时进行钱包转账和启动游戏是被禁止的。在这个情况，PG SOFT 将限制玩家操作并会对该钱包 API 请求返回错误代码：

```
1305: Invalid player - Player is locked: [playername]
```

错误处理

1305

若运营商收到此错误代码，需重新发送钱包转账请求。

PG SOFT 系统正在为所有 API 操作实现**幂等操作**。为了避免发生多个余额转账，要求运营商通过使用相同的 transfer_reference **重新发送**请求。

3004

若在创建玩家时发生无法处理的错误，PG SOFT 将返回此错误代码。运营商可通过 External API 重新创建玩家钱包或重新启动 PG SOFT 游戏以自动创建玩家。

1315

当遇到以下情况时，PG SOFT 将返回此错误代码：

- 并发动作（例如：该玩家同时使用两个不同的浏览器启动游戏等）。
- 玩家在进行交易的同时启动游戏或在游戏中进行旋转。
- 玩家钱包不存在。
- 在进行钱包转账时出现错误。

若运营商收到此错误代码，应在 5 分钟后重新发送钱包交易请求。

5.2.5 转出所有余额（任选项）

从特定玩家的现金钱包中转出所有钱包金额。

注解：

- Transfer_reference 是每笔交易的唯一识别码。运营商需要为每笔传送请求发送新的 transfer_reference。
- 对于重复请求，PG SOFT 不对玩家钱包余额进行任何操作，但返回转账成功的响应。
- 此 API 用于清空玩家的钱包余额。如若玩家没有任何余额时，系统将不会显示任何错误返回。运营商需要在 API 响应中检查玩家之前和当前的剩余金额，并且确认实际提款金额以避免余额丢失。

请求参数

API URL : {PgSoftAPIDomain}/Cash/v3/TransferAllOut
HTTP method : POST

URL 参数：

参数名	参数类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： 请把参数值设置为 GUID 格式

主体参数：

参数名	参数类型	必需项	描述
operator_token	String	是	运营商独有的身份识别
secret_key	String	是	PG SOFT 与运营商之间共享密码
player_name	String	是	玩家账号
transfer_reference	String	是	交易凭证，运营商平台生成并确保每次交易是独有的。 注：上限 50 字符
currency	String	是	玩家选择的币种

范例:

URL:

https://api.pg-bo.me/external/Cash/v3/TransferAllOut?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123&transfer_reference=ref1530879795000¤cy=EUR

返回结果

主体参数:

JSON 对象

参数名	参数类型	描述
transactionId	String	特殊交易识别码
balanceAmountBefore	Decimal	交易前的玩家余额
balanceAmount	Decimal	交易后的玩家余额
amount	Decimal	交易金额

范例:

```
{
  "data": {
    "transactionId": 12458848,
    "balanceAmountBefore": 47238.07,
    "balanceAmount": 0,
    "amount": 47238.07
  },
  "error": null
}
```

错误返回

代码	信息
3001	不能空值
3005	玩家钱包不存在

5.2.6 获取单个交易记录（任选项）

透过搜索交易凭证及玩家账号，可获取特定玩家的单个交易记录。

注解：

- 此 API 只能在最后一笔交易的 5 分钟后调用。

请求参数

API URL : {PgSoftAPIDomain}/Cash/v3/GetSingleTransaction
HTTP method : POST

URL 参数：

参数名	参数类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： 请把参数值设置为 GUID 格式

主体参数：

参数名	参数类型	必需项	描述
operator_token	String	是	运营商独有的身份识别
secret_key	String	是	PG SOFT 与运营商之间共享密码
player_name	String	是	玩家账号
transfer_reference	String	是	交易凭证，运营商平台生成并确保每次交易是独有的。 注：上限 50 字符

范例：

URL:

https://api.pg-bo.me/external/Cash/v3/GetSingleTransaction?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_name=player123&transfer_reference=ref1530879795000

返回结果

主体参数:

JSON 对象

参数名	参数类型	描述
transactionId	String	特殊交易识别码
playerName	String	玩家账号
currencyCode	String	玩家选择的币种
transactionType	Integer	交易类型: 100: TransferInCash 200: TransferOutCash
transactionAmount	Decimal	交易金额
transactionFrom	Decimal	交易前的钱包余额
transactionTo	Decimal	交易后的钱包余额
transactionDateTime	Long	交易的日期和时间 (Unix 时间戳, 以毫秒为单位)

范例:

```
{
  "data": {
    "transactionId": 35595943,
    "playerName": "player123",
    "currencyCode": "EUR",
    "transactionType": 100,
    "transactionAmount": 1000000.00000,
    "transactionFrom": 0.00,
    "transactionTo": 1000000.00,
    "transactionDateTime": 1522984865000
  },
  "error": null
}
```

错误返回

代码	信息
3040	交易不存在

5.2.7 投注详情页面（任选项）

显示玩家的投注结果和特定投注的详细信息。

步骤 1

在获取游戏记录详细图片之前，必须先获取运营商令牌。（限定时间：30 分钟）。

请求参数

API URL : {PgSoftAPIDomain}/Login/v1/LoginProxy

HTTP method : POST

URL 参数:

参数名	参数类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注: 请把参数值设置为 GUID 格式

主体参数:

参数名	参数类型	必需项	描述
operator_token	String	是	运营商独有的身份识别
secret_key	String	是	PG SOFT 与运营商之间共享密码

范例:

URL:

https://api.pg-bo.me/external/Login/v1/LoginProxy?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijk1

返回结果

主体参数:

JSON 对象

参数名	参数类型	描述
operator_session	String	运营商令牌

范例:

```
{
  "data": {
    "operator_session": E4ECB5CF-0BC6-4DA8-8551-8526F48AB9CD
  },
  "error": null
}
```

步骤 2

通过以下方式获取投注详情:

API URL : {PgSoftPublicDomain}/history/redirect.html

HTTP method : Get

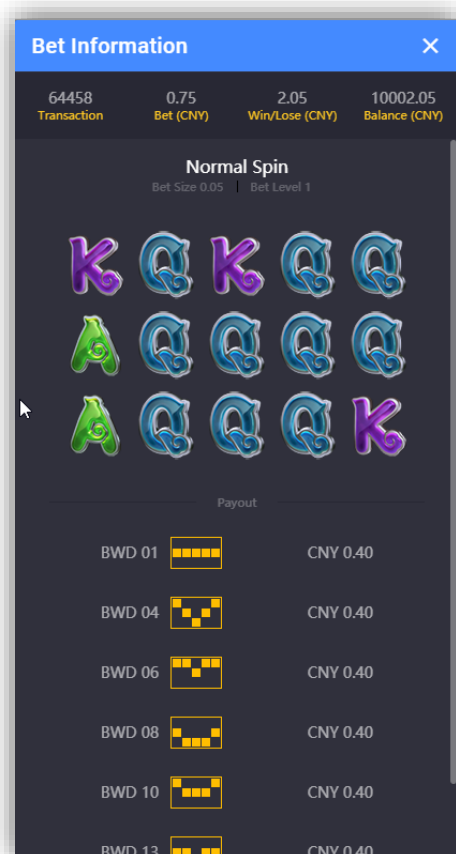
URL 参数:

参数名	参数类型	描述
trace_id	String	请求的唯一标识符 (GUID) 注: 请把参数值设置为 GUID 格式
t	String	步骤 1 的运营商令牌
psid	String	母注单 ID
sid	String	子投注 ID
gid	String	游戏 ID
lang	String	语言: • en: 英文 (默认) • zh: 中文
type	String	固定值: operator

范例:

https://public.pg-redirect.net/history/redirect.html?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311&t=E4ECB5CF-0BC6-4DA8-8551-8526F48AB9CD&psid=12345&sid=12345&gid=1&lang=en&type=operator

截图



游戏投注详情

5.2.8 获取多个玩家钱包余额（任选项）

获取多个玩家的钱包余额（最多 10 个玩家）。

注解：

- 最多可获取 10 个玩家的钱包余额。

请求参数

API URL : {PgSoftAPIDomain}/Cash/v3/GetPlayersWallet

HTTP method : POST

URL 参数：

参数名	参数类型	必需项	描述
trace_id	String	是	请求的唯一标识符（GUID） 注： 请把参数值设置为 GUID 格式

主体参数：

参数名	参数类型	必需项	描述
operator_token	String	是	运营商独有的身份识别
secret_key	String	是	PG SOFT 与运营商之间共享密码
player_names	String Array	是	玩家账号

范例：

URL:

https://api.pg-bo.me/external/Cash/v3/GetPlayersWallet?trace_id=b3f37e57-2873-40b1-aa95-f126c25ed311

主体:

operator_token=abc123&secret_key=a1b25cde5f3gh46ijkl&player_names=player1&player_names=player2

返回结果

主体参数:

JSON 对象

参数名	参数类型	描述
totalCount	Integer	记录总数
totalPage	Integer	可用批次的总数
result	JSON Array	玩家钱包详情列表

JSON 数组

参数名	参数类型	描述
playerName	String	玩家名称
currencyCode	String	玩家选择的币种
totalBalance	Decimal	玩家钱包余额

范例:

```
{
  "data": {
    "totalCount": 2,
    "totalPage": 1,
    "result": [
      {
        "playerName": "player1",
        "currencyCode": "CNY",
        "totalBalance": 33.59
      },
      {
        "playerName": "player2",
        "currencyCode": "CNY",
        "totalBalance": 2000061103.84
      }
    ]
  },
  "error": null
}
```

错误返回

代码	信息
3001	不能空值
3005	玩家钱包不存在

5.2.9 其他（任选项）

PG SOFT 为不同的模块提供了几个 API，可依据选择对接。运营商可根据需求进行对接：

- **投注历史记录** : 获取玩家历史记录。
- **游戏列表** : 获取最新游戏列表和可用的投注大小。
- **免费游戏** : 通过 API 执行多项操作（例如：创建免费游戏，给予玩家免费游戏，及取消免费游戏等）。
- **红利游戏** : 通过 API 执行多项操作（例如：创建红利游戏，给予玩家红利，及取消红利等）。
- **锦标赛** : 通过 API 将玩家添加到锦标赛中，并且获得锦标赛排名。

5.3 PG SOFT 游戏对接运营商 iOS 应用程序

运营商的应用程序需满足以下条件，才可在应用程序客户端中打开 PG SOFT 游戏

- iOS 8.0 或以上
- WKWebView

在设备未达到最低配置要求或是网络情况未达要求时，PG SOFT 游戏将出现“体验欠佳”的提示。玩家仍可继续进行游戏，但无法保证可获得最佳体验及稳定的 PG SOFT 游戏。

为保证应用程序客户端能通过 PG SOFT 游戏要求的检测，运营商需在初始化 WKWebView 组件时，添加以下代码：

```
WKUserContentController *controller = [[WKUserContentController alloc] init];  
[controller addScriptMessageHandler: self name: @"Could be any string value"];  
  
WKWebViewConfiguration *configuration = [[WKWebViewConfiguration alloc] init];  
configuration.userContentController = controller;  
  
WKWebView *webView = [[WKWebView alloc] initWithFrame: CGRectZero  
configuration:  
configuration];  
webView.uiDelegate = self;  
self.view = webView;
```

此操作是为了构造 WKUserContentController 对象，并在配置上设置其 userContentController 属性。请注意 userContentController 必须至少有一个 scriptMessageHandler（通过 addScriptMessageHandler 分配）。

更多关于 WKWebView 的详细说明，请参考 [Apple Developer Documentation](#)

6. 解决待处理投注

在某些情况下，玩家于游戏期间可能发生不可预见的情况，如网络延误、系统故障、系统超时等等。这将可能导致待处理投注的出现。

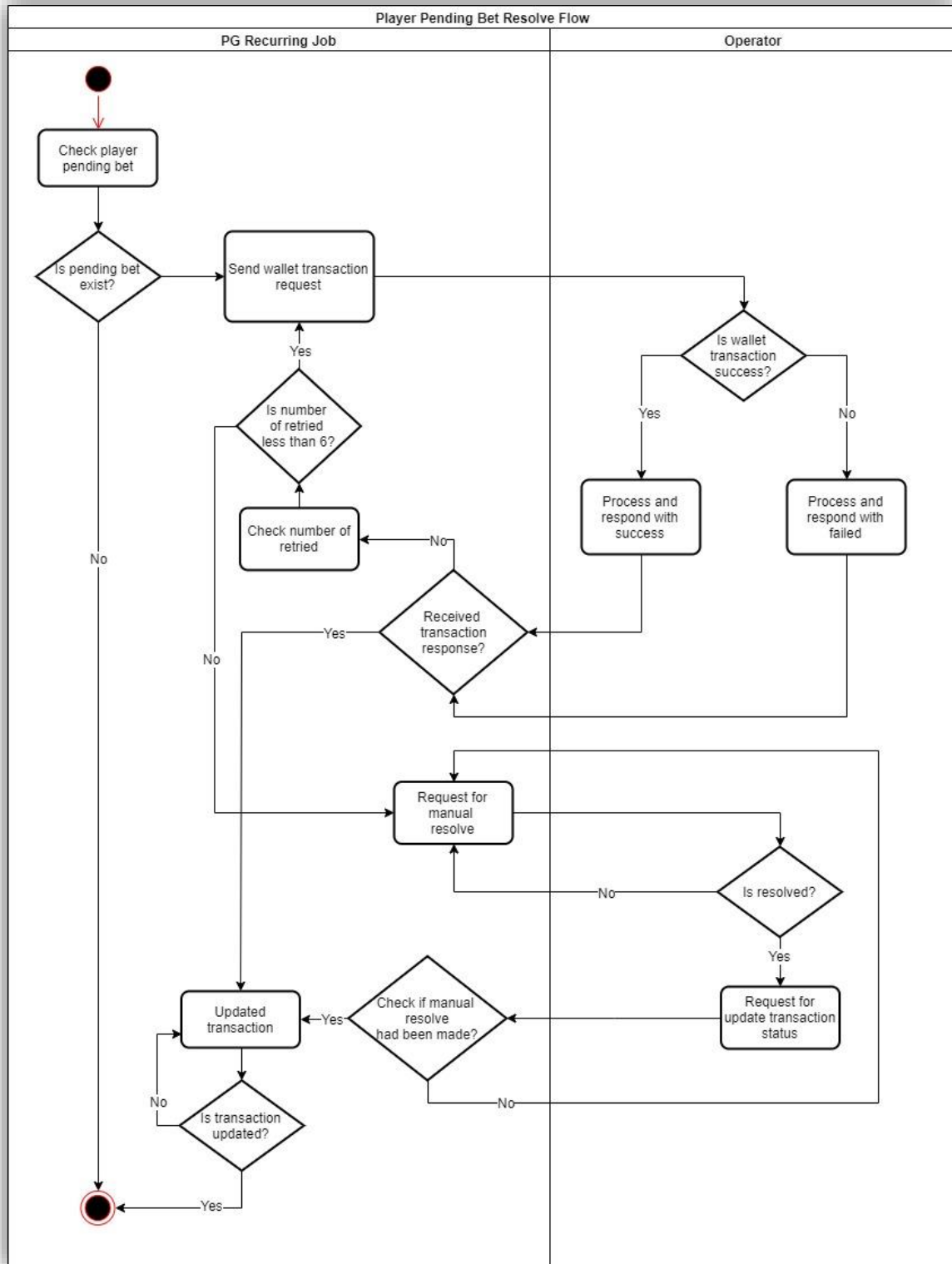
通常，当玩家重新启动游戏时，系统将自动解决该待处理投注。但是在某些罕见的情况下，投注将无法获得解决；并保持待处理状态：

- 当玩家遇到错误后，没有重新启动游戏；
- 解决待处理投注时，PG SOFT 系统未收到来自运营商的任何响应；
- 解决待处理投注时，PG SOFT 系统会收到来自运营商的错误响应。

解决待处理投注的默认间隔时间设置为 60 分钟。超过间隔的待处理投注将由 PG SOFT 系统自动解决。该间隔是可自定义的，每个运营商都可自定义解决待处理投注的间隔时间。

若尝试几次后仍无法解决待处理投注，则需要手动解决。在这种情况下，PG SOFT 将要求运营商手动解决待处理投注。在获得运营商确认后，PG SOFT 将会手动把该投注的状态标记为完成。

截图



玩家待处理投注的解决流程

7. 重置游戏状态

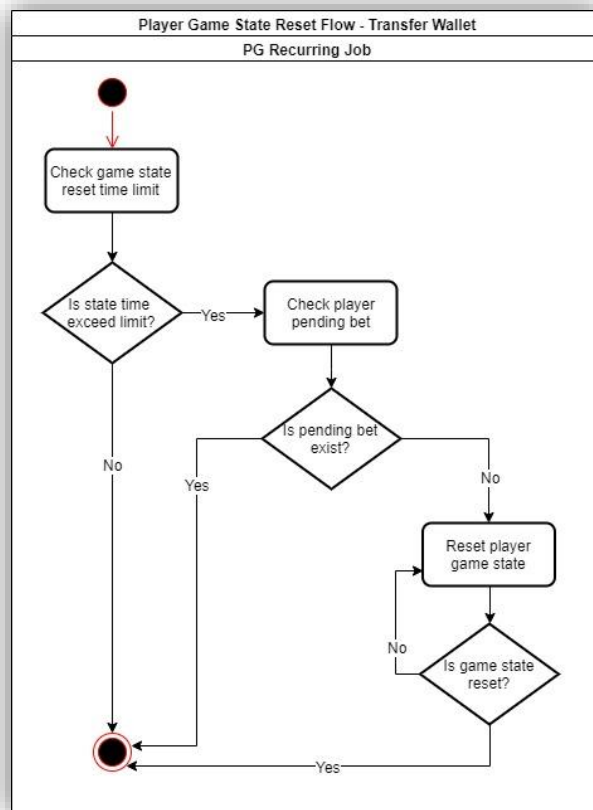
所有游戏都支持恢复游戏状态的功能，该功能允许玩家在任何设备回到上一个游戏状态并继续游戏。举例，玩家在赢得免费旋转后关闭了游戏，但玩家依然可以在几天后继续进行免费旋转。

但是，默认情况下，每 90 天重置玩家的游戏状态。此间隔是可自定义的，每个运营商都可自定义游戏状态重置的间隔时间。系统将清除所有已完成和未完成的游戏进度，并且玩家在重置游戏后将无法继续游戏。

对于转账钱包模式，默认情况下每 90 天将重置每个玩家的游戏状态：

- 若无任何待处理投注，PG SOFT 系统将自动重置玩家的游戏状态。

截图



转账钱包的玩家游戏状态重置流程

8. 限制

8.1 IP 限制

来自以下限制区域的玩家将无法访问 PG SOFT 游戏：

- 马来西亚
- 新加坡
- 台湾
- 美国
- 香港
- 澳门
- 以色列
- 伊朗
- 北韓
- 澳大利亚

8.2 货币限制

以下为不被 PG SOFT 所支持的货币：

- AUD
- ESP
- HRK
- STD
- TRL
- HKD
- MOP
- MYR
- SGD
- TWD
- VES
- MBTC, UBTC, USDT, TUSD & USDC 除外的所有加密货币

9. 附录

9.1 错误代码

错误代码	描述
1034	无效请求
1035	行动失败
1200	内部服务器失败
1204	无效运营商
1300	无效玩家令牌
1301	玩家令牌空置
1302	无效玩家令牌
1303	发生服务器错误
1305	无效玩家
1306	玩家被阻止访问当前游戏
1307	无效玩家令牌
1308	玩家令牌已过期
1309	玩家已被封锁
1310	运营商和玩家令牌验证失败
1315	玩家操作进行中
1400	游戏正在维护中
1401	游戏无效
1402	游戏不存在
3001	不能空值
3004	玩家不存在
3005	玩家钱包不存在
3006	玩家钱包已存在
3009	免费游戏不存在
3013	余额不足无法转出

3014	免费游戏无法取消
3019	免费游戏不足
3021	投注不存在
3022	投注已支付
3030	免费游戏过期
3031	免费游戏已经转换
3032	投注已存在
3033	投注失败
3034	支付失败
3035	倍数错误
3036	余额不足无法转换
3040	交易不存在
3202	余额不足无法投注

9.2 货币代码

货币代码	货币名称	基础单位
AED	阿拉伯联合酋长国迪拉姆	1
AFN	阿富汗	1
ALL	阿尔巴尼亚列克	1
AMD	亚美尼亚德拉姆	1
ANG	安的列斯盾	1
AOA	安哥拉宽扎	1
ARS	阿根廷比索	1
AWG	阿鲁巴弗罗林	1
AZN	阿塞拜疆马纳特	1
BAM	波黑可兑换马克	1
BBD	巴巴多斯元	1
BDT	孟加拉塔卡	1
BGN	保加利亚列弗	1
BHD	巴林第纳尔	1
BIF	布隆迪法郎	1000
BMD	百慕大元	1
BND	汶莱元	1
BOB	玻利维亚诺	1
BRL	巴西里亚伊（雷亚尔）	1
BSD	巴哈马元	1
BTN	不丹努尔特鲁姆	1
BWP	博茨瓦纳普拉	1
BYN	白俄罗斯卢布	1
BYR	白俄罗斯卢布	1
BZD	伯利兹元	1
CAD	加拿大元	1

CDF	刚果法郎	1000
CHF	瑞士法郎	1
CLP	智利比索	1
CNY	人民币	1
COP	哥伦比亚比索	1000
CRC	哥斯达黎加科朗	1
CSD	塞尔维亚第纳尔	1
CUP	古巴比索	1
CVE	佛得角埃斯库多	1
CZK	捷克克朗	1
DJF	吉布提法郎	1
DKK	丹麦克朗	1
DOP	多米尼加比索	1
DZD	阿尔及利亚第纳尔	1
EGP	埃及镑	1
ERN	厄立特里亚纳克法	1
ETB	埃塞俄比亚比尔	1
EUR	欧元	1
FJD	斐济元	1
FKP	福克兰岛磅	1
GBP	英镑	1
GEL	格鲁吉亚拉里	1
GHS	加纳塞地	1
GIP	直布罗陀庞德	1
GMD	冈比亚货币	1
GNF	几内亚法郎	1000
GTQ	危地马拉格查尔	1
GYD	圭亚那元	1
HNL	洪都拉斯伦皮拉	1

HTG	海地古德	1
HUF	匈牙利福林	1
IDR	印度尼西亚卢比盾	1000
ILS	以色列谢克尔	1
INR	印度卢比	1
IQD	伊拉克第纳尔	1000
IRR	伊朗里亚尔	1000
ISK	冰岛克朗	1
JMD	牙买加元	1
JOD	约旦第纳尔	1
JPY	日元	1
KES	肯尼亚先令	1
KGS	吉尔吉斯斯坦索姆	1
KHR	柬埔寨利尔斯	1000
KMF	科摩罗法郎	1
KPW	北朝鲜元	1
KRW	韩元	1000
KWD	科威特第纳尔	1
KYD	开曼岛元	1
KZT	坚戈	1
LAK	老挝基普	1000
LBP	黎巴嫩镑	1000
LKR	斯里兰卡卢比	1
LRD	黎巴嫩元	1
LSL	莱索托洛蒂	1
LVL	拉脱维亚拉特	1
LYD	利比亚第纳尔	1
MAD	摩洛哥迪拉姆	1
MBTC	比特币（虚拟货币）	1

MDL	摩尔多瓦列伊	1
MGA	马达加斯加阿里亚里	1000
MKD	马其顿第纳尔	1
MMK	缅甸元	1000
MNT	蒙古圖格裡克	1000
MUR	毛里求斯卢比	1
MVR	马尔代夫罗非亚	1
MWK	马拉维克瓦查	1
MXN	墨西哥比索	1
MZN	莫桑比克梅蒂卡尔	1
NAD	纳米比亚元	1
NGN	尼日利亚奈拉	1
NIO	尼加拉瓜科多巴	1
NOK	挪威克朗	1
NPR	尼泊尔卢比	1
NZD	新西兰元	1
OMR	阿曼里亚尔	1
PAB	巴拿马巴波亚	1
PEN	秘鲁索尔	1
PGK	巴布亚新几内亚基纳	1
PHP	菲律宾披索	1
PKR	巴基斯坦卢比	1
PLN	波兰兹罗提	1
PYG	巴拉圭瓜拉尼	1000
QAR	卡塔尔里亚尔	1
RON	罗马尼亚列伊	1
RSD	塞尔维亚第纳尔	1
RUB	俄罗斯卢布	1
RWF	卢旺达法郎	1000

SAR	沙特里亚尔	1
SBD	所罗门群岛元	1
SCR	塞舌尔卢比	1
SDG	苏丹镑	1
SEK	瑞典克朗	1
SHP	圣赫勒拿磅	1
SLL	塞拉利昂利昂	1000
SOS	索马里先令	1
SRD	苏里南元	1
SVC	萨尔瓦多科朗	1
SYP	叙利亚镑	1
SZL	斯威士马兰吉尼	1
THB	泰铢	1
TJS	塔吉克斯坦索莫尼	1
TMT	土库曼斯坦新马纳特	1
TND	突尼斯第纳尔	1
TOP	汤加潘加	1
TRY	土耳其里拉	1
TTD	特立尼达与多巴哥元	1
TUSD	TrueUSD (虚拟货币)	1
TZS	坦桑尼亚先令	1000
UAH	乌克兰赫夫米	1
UBTC	微比特币	1
UGX	乌干达先令	1000
USD	美元	1
USDC	USD Coin (虚拟货币)	1
USDT	泰达币	1
UYU	乌拉圭比索	1
UZS	乌兹别克斯坦苏姆	1000

VND	越南盾	1000
VUV	瓦努阿图瓦图	1
WST	萨摩亚塔拉	1
XAF	中非金融合作法郎	1
XCD	东加勒比元	1
XOF	CFA 法郎	1
XPF	太平洋法郎	1
YER	也门里亚尔	1
ZAR	南非兰特	1
ZMW	赞比亚克瓦查	1

9.3 投注类型

投注类型	游戏模式
1	真钱模式
3	锦标赛模式

9.4 交易类型

代码	交易类型
1	现金
2	红利游戏
3	免费游戏

9.5 平台

代码	类型	平台
1	网页端	Windows
2		macOS
3		Android
4		iOS
5		其他
6	Cordova	Android
7		iOS
8	Electron	Windows
9		macOS
10	原生应用	Windows
11		macOS
12		Android
13		iOS
98	其他	系统
99		挂机

9.6 语言

代码	语言
en	英文（默认）
da	丹麦文
de	德文
es	西班牙文
fi	芬兰文
fr	法文
id	印尼文
it	意大利文
ja	日文
ko	韩文
nl	荷兰文
no	挪威文
pl	波兰文
pt	葡萄牙文
ro	罗马尼亚文
ru	俄文
sv	瑞典文
th	泰文
tr	土耳其文
vi	越南文
zh	中文
my	缅甸文