

BITGAMEX 小游戏接入文档

目录

1. 版本日志	4
2. 交互示意	5
3. 结算脚本函数	5
4. 登录接口	6
4.1. 接口说明	6
4.2. 接口参数	6
4.3. 响应结果	7
4.4. 特殊说明	8
5. 存盘数据获取接口	8
5.1. 接口说明	8
5.2. 接口参数	8
5.3. 响应结果	8
6. 存盘接口	9
6.1. 接口说明	9
6.2. 接口参数	9
6.3. 响应结果	10
6.4. 特殊说明	11
7. 转账游戏币给其他玩家的接口	11
7.1. 接口说明	11
7.2. 接口参数	11
7.3. 响应结果	12
7.4. 特殊说明	12
8. 绑定 BIT.GAME 交易所账号的接口	13
8.1. 接口说明	13
8.2. 接口参数	13
8.3. 响应结果	13
8.4. 特殊说明	14
9. 转账游戏币给绑定的交易所账号的接口	14

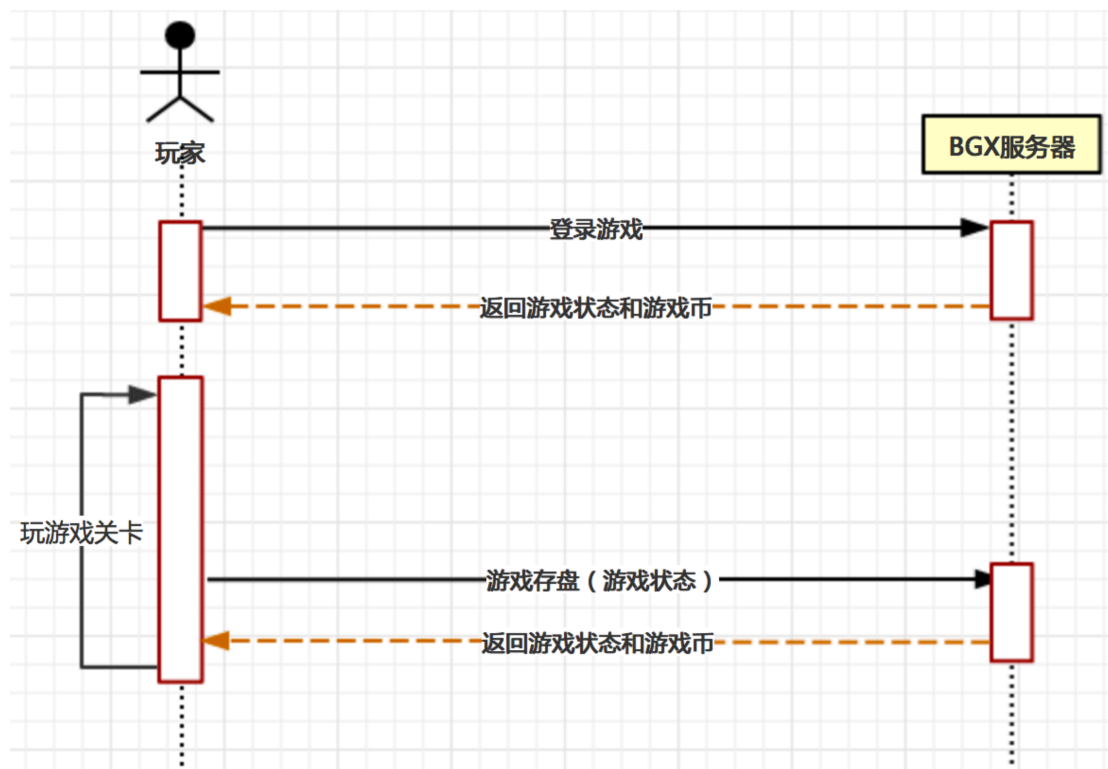
9.1. 接口说明	14
9.2. 接口参数	15
9.3. 响应结果	15
9.4. 特殊说明	16
10. 绑定以太坊钱包地址的接口	16
10.1. 接口说明	16
10.2. 接口参数	16
10.3. 响应结果	17
10.4. 特殊说明	18
11. 转账游戏币给绑定的钱包的接口	18
11.1. 接口说明	18
11.2. 接口参数	18
11.3. 响应结果	19
11.4. 特殊说明	19

1. 版本日志

修改记录

版本号	修改日期	变更说明	修改人	审核	备注
0.1	2018-3-2	起草	谷力		
0.2	2018-3-5	增加转账相关接口	谷力		

2. 交互示意



3. 结算脚本函数

为方便控制和修改游戏币的产出，每个关卡结束时游戏币的结算将放在 BGX 游戏服务器上计算，因此，游戏研发方需要根据游戏自身的关卡奖励逻辑提供一个 Lua 脚本函数 F 来计算玩家的游戏币增量，假设关卡开始时玩家游戏状态的数据结构为 S0，游戏币余额为 B0，结束时状态变化成 S，则：

游戏币增量 = $F(S0, S)$

说明：最终的游戏币余额，BGX 服务器会在 B0+增量的基础上进行额外规则的调整。存盘接口的返回结果中将提供 F 的结果，方便联调。

4. 登录接口

4.1. 接口说明

使用场景：游戏客户端请求 BGX 游戏服务器登录某游戏

接口使用者：游戏客户端以 HTTPS GET 方式请求。

请求地址：https://api.bitgamex.com/?a=login_game

4.2. 接口参数

参数	字段类型	必填	说明
uid	Int(11)	是	用户id
game_id	Int(11)	是	游戏的标识
device_id	String(200)	是	设备id
time	Int(20)	是	时间戳
device_model	String(32)	否	设备型号
os_type	String(20)	否	操作系统类型
os_ver	String(100)	否	操作系统版本
lang	String(50)	否	语言
org_device_id	String(200)	否	最初的设备id
gc_id	String(40)	否	绑定ios game center账号id
gg_id	String(40)	否	绑定google账号id
fb_id	String(40)	否	绑定facebook账号id
sign	String(32)	是	MD5签名 sign = md5(uid+game_id+device_id+ time+key) “+”表示直接将两个字符串连接，拼接值不要有空格。用标准MD5算法进行加密。其中

			key将在对接过程中由双方一同确定，比如： "BIT.GAME.X.8.8.8.8"
--	--	--	---

4.3. 响应结果

类型：json 结构

成功：

```
{
  "succ":1,
  "uid":Int(11),
  "token":String(50), 玩家在游戏中用于存盘的令牌
  "game_data":String(2048), base64 编码，玩家该游戏的存盘数据
  "balance":Double, 玩家的游戏币余额
}
```

失败：

```
{
  "succ":0,
  "req":"login_game",
  "errno":错误编号
  "errmsg":错误描述
}
```

errno值	说明
-999	ip受限
-998	参数不全
-997	参数错误
-996	校验失败
-995	协议不支持
-994	异常抛出
-993	游戏未开放
-992	读缓存失败
-991	不存在rpc服务器

-990	金币不足
-989	请求失败
-988	请求超时
-987	未确定的错误

* 以上为公共错误号定义，也为其他接口共享，后面不再重复列出。各接口特定的错误号会重复定义，结合错误返回结果中的 **req** 字段可唯一确定。

4.4. 特殊说明

该接口在玩家进入某游戏时调用，用于获取该游戏的存盘数据和令牌（后面存盘校验用）。对该接口的再次调用，将会默认玩家从上次进入的游戏中退出，上次的令牌失效。

5. 存盘数据获取接口

5.1. 接口说明

使用场景：游戏客户端请求 BGX 游戏服务器下发玩家在游戏时的存盘数据，以及游戏币余额。

接口使用者：游戏客户端以 HTTPS GET 方式请求。

请求地址：https://api.bitgamex.com/?a=get_game

5.2. 接口参数

参数	字段类型	必填	说明
uid	Int(11)	是	用户id
game_id	Int(11)	是	游戏的标识
token	String(50)	是	在登录接口中返回的令牌

5.3. 响应结果

类型: json 结构

成功:

```
{
  "succ":1,
  "game_data":String(2048), base64 编码, 玩家该游戏的存盘数据
  "balance":Double, 玩家的游戏币余额
}
```

失败:

```
{
  "succ":0,
  "req":"get_game",
  "errno":错误编号
  "errmsg":错误描述
}
```

errno值	说明

6. 存盘接口

6.1. 接口说明

使用场景: 游戏客户端请求 BGX 游戏服务器存盘玩家的游戏状态数据, 并获得新的游戏币余额。

接口使用者: 游戏客户端以 HTTPS POST 方式请求。

请求地址: https://api.bitgamex.com/?a=save_game

6.2. 接口参数

参数	字段类型	必填	说明
uid	Int(11)	是	用户id
game_id	Int(11)	是	游戏的标识
token	String(50)	是	在登录接口中返回的令牌
game_data	String(2048)	是	该游戏的状态数据
time	Int(20)	是	时间戳
sign	String(32)	是	MD5签名 sign = md5(uid+game_id+token+game_data+time+key) “+”表示直接将两个字符串连接，拼接值不要有空格。用标准MD5算法进行加密。其中key和登录接口中一致。

6.3. 响应结果

类型：json 结构

成功：

```
{
  "succ":1,
  "game_data":String(2048), base64 编码，玩家该游戏的存盘数据
  "balance":Double, 玩家的游戏币余额
  "f_res":Double, 玩家的游戏币基础增量，方便联调检查
}
```

失败：

```
{
  "succ":0,
  "req":"save_game",
  "errno":错误编号
  "errmsg":错误描述
}
```

errno值	说明

6.4. 特殊说明

表单参数中的 time 字段用于存盘时间校验。

7. 转账游戏币给其他玩家的接口

7.1. 接口说明

使用场景：游戏客户端请求 BGX 游戏服务器将自己的一部分游戏币转账给另一玩家。

接口使用者：游戏客户端以 HTTPS GET 方式请求。

请求地址：https://api.bitgamex.com/?a=transfer_coin_in_game

7.2. 接口参数

参数	字段类型	必填	说明
uid	Int(11)	是	用户id
game_id	Int(11)	是	游戏的标识
token	String(50)	是	在登录接口中返回的令牌
dst_uid	Int(11)	是	目标用户id
amount	Double	是	转账的游戏币数量
time	Int(20)	是	时间戳
sign	String(32)	是	MD5签名 sign = md5(uid+game_id+token+dst_

			uid+amount+time+key) “+”表示直接将两个字符串连接，拼接值不要有空格。用标准MD5算法进行加密。其中key将在对接过程中由双方一同确定，比如：“BIT.GAME.X.8.8.8.8”
--	--	--	---

7.3. 响应结果

类型：json 结构

成功：

```
{
  "succ":1,
  "balance":Double, 玩家的游戏币余额
}
```

失败：

```
{
  "succ":0,
  "req":"transfer_coin_in_game",
  "errno":错误编号
  "errmsg":错误描述
}
```

errno值	说明
-1	目标玩家错误（自己，或者不存在）

7.4. 特殊说明

转账会收取一定的手续费。

8. 绑定 BIT.GAME 交易所账号的接口

8.1. 接口说明

使用场景：游戏客户端请求 BGX 游戏服务器将指定的 BIT.GAME 交易所的账号绑定到自己的用户 id 上，方便以后转账到交易所。

接口使用者：游戏客户端以 HTTPS GET 方式请求。

请求地址：https://api.bitgamex.com/?a=bind_exchange_accid

8.2. 接口参数

参数	字段类型	必填	说明
uid	Int(11)	是	用户id
game_id	Int(11)	是	游戏的标识
token	String(50)	是	在登录接口中返回的令牌
exchange_accid	String(100)	是	交易所账号id
time	Int(20)	是	时间戳
sign	String(32)	是	MD5签名 sign = md5(uid+game_id+token+exchange_accid+time+key) “+”表示直接将两个字符串连接，拼接值不要有空格。用标准MD5算法进行加密。其中key将在对接过程中由双方一同确定，比如：“BIT.GAME.X.8.8.8.8”

8.3. 响应结果

类型：json 结构

成功：

```
{
  "succ":1,
  "exchange_accid":String(100), 指定的交易所账号 id
}
```

失败：

```
{
  "succ":0,
  "req":"bind_exchange_accid",
  "errno":错误编号
  "errmsg":错误描述
}
```

errno值	说明

8.4. 特殊说明

玩家自己对所指定的交易所账号的正确性负责。

如果再次调用此接口，将覆盖原来绑定的交易所账号。

9. 转账游戏币给绑定的交易所账号的接口

9.1. 接口说明

使用场景：游戏客户端请求 BGX 游戏服务器将自己的一部分游戏币转账给绑定的交易所账号。

接口使用者：游戏客户端以 HTTPS GET 方式请求。

请求地址: https://api.bitgamex.com/?a=transfer_coin_to_exchange

9.2. 接口参数

参数	字段类型	必填	说明
uid	Int(11)	是	用户id
game_id	Int(11)	是	游戏的标识
token	String(50)	是	在登录接口中返回的令牌
amount	Double	是	转账的游戏币数量
time	Int(20)	是	时间戳
sign	String(32)	是	MD5签名 sign = md5(uid+game_id+token+amount+time+key) “+”表示直接将两个字符串连接, 拼接值不要有空格。用标准MD5算法进行加密。其中key将在对接过程中由双方一同确定, 比如: "BIT.GAME.X.8.8.8.8"

9.3. 响应结果

类型: json 结构

成功:

```
{
  "succ":1,
  "balance":Double, 玩家在游戏游戏中的游戏币余额
  "exchange_balance":Double, 玩家绑定的交易所账号中的游戏币余额
}
```

失败:

```
{
```

```

    "succ":0,
    "req":"transfer_coin_to_exchange",
    "errno":错误编号
    "errmsg":错误描述
}

```

errno值	说明

9.4. 特殊说明

转账会收取一定的手续费。

10. 绑定以太坊钱包地址的接口

10.1.接口说明

使用场景：游戏客户端请求 BGX 游戏服务器将指定的以太坊钱包的地址绑定到自己的用户 id 上，方便以后转账到钱包。

接口使用者：游戏客户端以 HTTPS GET 方式请求。

请求地址：https://api.bitgamex.com/?a=bind_wallet

10.2.接口参数

参数	字段类型	必填	说明
uid	Int(11)	是	用户id
game_id	Int(11)	是	游戏的标识
token	String(50)	是	在登录接口中返回的令牌
wallet_addr	String(50)	是	以太坊钱包地址

time	Int(20)	是	时间戳
sign	String(32)	是	MD5签名 sign = md5(uid+game_id+token+wallet_addr+time+key) “+”表示直接将两个字符串连接，拼接值不要有空格。用标准MD5算法进行加密。其中key将在对接过程中由双方一同确定，比如：“BIT.GAME.X.8.8.8.8”

10.3.响应结果

类型：json 结构

成功：

```
{
  "succ":1,
  "wallet_addr":String(50), 指定的以太坊钱包地址
}
```

失败：

```
{
  "succ":0,
  "req":"bind_wallet",
  "errno":错误编号
  "errmsg":错误描述
}
```

errno值	说明

10.4.特殊说明

玩家自己对所指定的以太坊钱包地址的正确性负责。

如果再次调用此接口，将覆盖原来绑定的钱包地址。

11. 转账游戏币给绑定的钱包的接口

11.1.接口说明

使用场景：游戏客户端请求 BGX 游戏服务器将自己的一部分游戏币转账给绑定的钱包。

接口使用者：游戏客户端以 HTTPS GET 方式请求。

请求地址：https://api.bitgamex.com/?a=transfer_coin_to_wallet

11.2.接口参数

参数	字段类型	必填	说明
uid	Int(11)	是	用户id
game_id	Int(11)	是	游戏的标识
token	String(50)	是	在登录接口中返回的令牌
amount	Double	是	转账的游戏币数量
time	Int(20)	是	时间戳
sign	String(32)	是	MD5签名 sign = md5(uid+game_id+token+amount+time+key) “+”表示直接将两个字符串连接，拼接值不要有空格。用标准MD5算法进行加密。其中key将在对接过程中由双方一

			同确定，比如： "BIT.GAME.X.8.8.8.8"
--	--	--	------------------------------

11.3.响应结果

类型：json 结构

成功：

```
{
  "succ":1,
  "balance":Double, 玩家在游戏游戏中的游戏币余额
  "exchange_balance":Double, 玩家绑定的交易所账号中的游戏币余额
}
```

失败：

```
{
  "succ":0,
  "req":"transfer_coin_to_wallet",
  "errno":错误编号
  "errmsg":错误描述
}
```

errno值	说明

11.4.特殊说明

转账会收取双重的手续费（1. 到交易所 2. 交易所到钱包）。