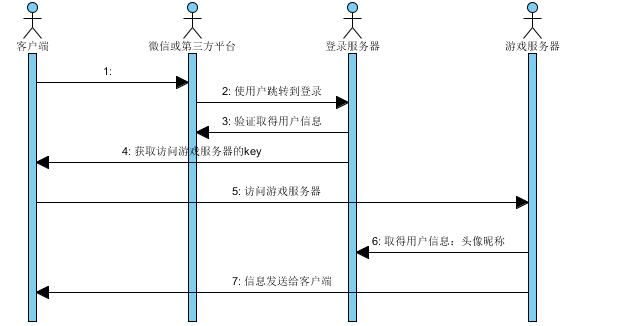
## 服务器文档

一.用户登录过程如下图：



二.登录服务器介绍:

登录服务器功能

a. 记录信息 1.用户帐号信息 2.登录动作信息 3.订单信息 4.平台信息

5.服务器列表信息 7.游戏指向信息8.游戏商品信息 9.游戏公告信息

b. 根据定位条件定位指定的游戏服务。

c. 用户的账单校对处理。

d.通知游服给玩家发放指定的充值物品。

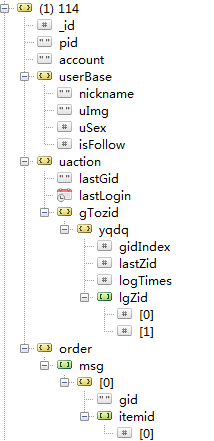
三.数据实体介绍：

用户信息实体：UserInfo：

userBase 用户基本信息

uaction用户登录过最后登录过的游戏 以及 登录过的区 未登录该节点不存在

order 用户支付动作记录（只记录支付的是那个游戏，支付过那个档位的商品。）

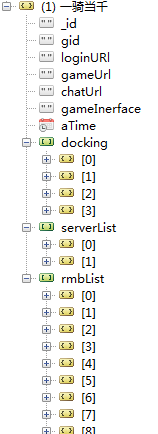
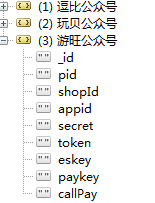


游戏平台实体：PlatformInfo

docking当前游戏对接平台信息

serverList 当前游戏开服列表 微信平台实体WxPlatform

rmbList 当前游戏的商品列表 微信相关信息字段

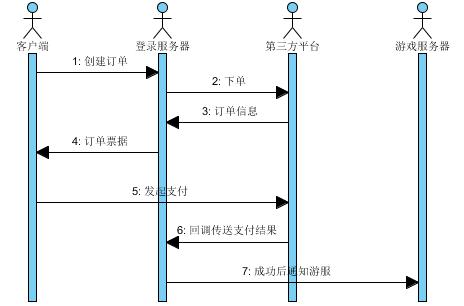
四.登录服对接第三方平台：

A. 根据不同平台需求编写SDK传到服务器Lib/sdk/下可立即生效。

一般平台对接主要有两项也是比不可少的两项：

1.登录用户的校验获取用户基本信息。

2.用户发起支付，到支付完成。下图：（只有正常流程）



3.订单详细记录过程数据在mysql数据库中请参看实体**Paymement ORM为mybiats**

B.SDK编写举例：Map<String, Object> param是http发过来的所有参数

***@Service***("SdkWxImpl")【生成bean的名字：规则Sdk程序写死｛（Wx）平台名称简写｝（Impl程序写死）】

***@Scope***("prototype")

**public** **class** **SdkWxImpl** **implements** I\_Platform｛

***@Override // 用户登录校验接口***

gid ,pid 对接连接专用。如果对接第三方平台有相同参数，param 中的键值将为g\_id,或p\_id,用这两个转换参数取第三方平台发过来的值。

**public** **UserBase** **logVerification**(Map<String, Object> param, **DataConf** dSource) **throws** **Exception** {

// 访问微信服务器获取用户信息(dSource.wx 微信平台配置信息，微信独用)

// (dSource.doc 其他平台对接信息)

**WxUser** **wUser** =wxservice.getUserWeb((**String**)param.get("code"),dSource.wx);

// 用户对象赋值

**if** (wUser != **null** && wUser.openid == **null**)

**return** **null**;

**UserBase** **ubase** = **new** UserBase();

dSource.rtn.getInfo().setAccount(wUser.openid);// 接收微信帐号

ubase.setuSex(wUser.sex);

ubase.setNickname(wUser.nickname);

ubase.setuImg(wUser.headimgurl);

dSource.rtn.setUsebase(ubase);

**return** ubase;｝

填充UserBase 并返回；

***@Override //创建订单返回订单信息***

**public** **Object** **payVerification**(Map<String, Object> param, **DataConf** dSource) **throws** **Exception** {

**String** **xml** = wxservice.uintOrder(param, dSource);

Map<String, String> **result** = **WxLogic**.*parseXml*(xml);

**WxResult** **wxres** = **new** WxResult();

wxres.setAppid(result.get("appid"));

wxres.setResult\_code(result.get("return\_code"));

wxres.setNonce\_str(result.get("nonce\_str"));

wxres.setPrepay\_id(result.get("prepay\_id"));

wxres.setTimeStamp(**WxLogic**.*getTimeStamp*());

wxres.setTrade\_type("MD5");

wxres.setReturn\_msg(dSource.wx.getShopId());

**WxLogic**.*launchOrder*(wxres, dSource.wx.getPaykey());

**return** wxres;

｝ //返回订单内容（这个是根据客户端需求返回）详细请看代码。

五.服务器部署

1.登录服务器 （同时并发为500，人流多可部署多个）



2.游戏服务器部署

a.大区部署：可采用上图登录服务器的部署方式。

b.滚服部署:只需填写登录服务器中的服务器列表即可。