需求描述文档

# 积分类型

积分用于衡量一个玩家在线的时间

## 普通积分

普通积分，适用于新玩家在不熟悉游戏的情况下使用，该类积分不具有换道具以及其他功能，仅仅是对新手锻炼经验之用，对新玩家每天采取送分的策略，分数用完后，立即送分，每天送指定次数的分数。

送分规则：

* 仅仅积分类型送分；
* 每天送10次分数；
* 游戏分数不够的时候，送分并通知玩家有送分
* 一次输出的分数超过了一次的送分分数，需要保障玩家不被锁定。
* 送分的过程与具体的游戏无关，要在游戏服务器中实现

送分流程（服务器端）：

坐下

基础分数

游戏开始

不够

送分次数

不够

结束

执行送分

结算分数

登录

送分过程：

在数据库表中记录送分次数，用户登录后，需要读取数据库中的送分次数（送分次数用完后，以晚上零点为界限），然后记录到服务器用户列表项中，在用户离开写入分数之前，不读写送分次数数据库，也就是需要在服务器用户列表项中维护送分次数

1. 用户登录，是第一次进入，服务器用户列表中没有改用户，读取数据库构造服务器用户列表数据，记录当前剩余的送分次数，记录当天是否已经送分
2. 用户登录房间成功后，

在服务器端:检查用户的当前分数和当前送分次数

1. 分数不够，送分次数不够，通知大厅客户端“你的分数不够继续游戏，你今天的送分次数已经用完”。
2. 分数足够，送分次数不够，不通知大厅客户端消息。
3. 分数不够，送分次数够，通知大厅客户端“你的分数不够继续游戏，今天第N次送分XXX分，还剩余送分次数M，欢迎继续游戏”。
4. 分数不够，送分次数不够，通知大厅客户端“你的分数不够继续游戏，你没有送分次数，只能旁观游戏”。
5. 游戏客户端：执行坐下动作，检查用户是否有权限坐下，检验用户分数是否符合设置的要求
6. 游戏结束后

服务器端：结算分数和送分次数，判断用户输分是否超过手上已有的分数

1. 分数不够，送分次数不够，发送弹出游戏命令，发送大厅信息“你分数不够，今天的送分次数已经用完，不能继续游戏”。
2. 分数不够，送分次数够，送分并更新用户状态，在聊天窗口中发送消息，“提示送给MM用户N分数”，发送当前送分用户信息，游戏窗口弹出提示“这是今天第N次送M分，今天的剩余送分次数为K”。
3. 分数够，送分次数不够，什么都不做。
4. 分数够，送分次数够，什么都不做。

登录

坐下?

READY?

弹出

READY

分数不够，并且

送分次数不够（非旁观）

分数够，送分次数够

不送分

判断分数不够，送分次数够，送分，等待游戏开始

在响应SUB\_GF\_USER\_READY消息的时候，通过m\_pITableUserAction->OnActionUserReady(pUserData->wChairID,m\_pIUserItem[pUserData->wChairID],(VOID \*)pDataBuffer,wDataSize);调用具体的游戏服务器钩子实现方法

CTableFrameSink::OnActionUserReady ，在其中添加游戏送分规则

服务器端

客户端

坐下请求

核对送分次数和分数发送通知

开始游戏或者弹出用户

游戏过程

游戏结束，分数结算核对

## 货币积分

该类型积分，可用于购买道具、参加比赛、等多种活动；

# 非比赛

非比赛区，根据底分和番数，设置有不同的房间，根据玩家的情况进行，不同的房间，番数和底数都是不同的。但是货币积分区和普通积分区是两种不同的记录方式。

## 普通积分游戏

普通游戏积分，仅仅作为玩家经验的增长

## 货币积分游戏

玩家赢得货币积分后，可以购买道具，可以购买装备，可以参加比赛

# 比赛

为了增加游戏的乐趣，设置比赛区域，玩家参加比赛，需要使用货币积分报名参赛，报名费作为比赛的奖品返回给赢家。

比赛是否需要设置多个标准区？例如，4番满，5番满，底分50，底分100的各种组合？

## 比赛规则

* 比赛使用比赛专用积分，不能转化为普通积分和货币积分，仅仅作为游戏比赛的过程中统计分数使用。
* 比赛不限局数目，在指定的比赛时间结束后，按照比赛专用积分进行排名；
* 比赛过程中，玩家查看不到其他人的信息，防止其他玩家作弊；
* 对逃跑超过三次的玩家将取消参赛资格；
* 比赛分数可以出现负分，不影响最后统计结果，但是参加比赛必须要有货币积分作为参赛的报名费；

## 比赛报名

* 不需要特别的报名，只要进入比赛场参加比赛，统计比赛期间的得分即可；
* 参加比赛，首先检验玩家是否有足够的报名费。

## 比赛时间

比赛为循环时间，每2天开始一场比赛，玩家可以在在这2天中任何时候参加都比赛

## 比赛评奖

对奖品设置按照比赛积分排序，在网站上进行公示。设置月冠军、年度冠军，对月冠军、年度冠军进行特殊奖励

# 游戏流程

打开程序

登陆

进入房间

选择座子坐下

分数检验

坐下成功

送分次数

不足

离开

不足

送分

足够

开始

结束结算

分数检验

不足

足够

登录失败

禁止游戏

不能坐下

# 游戏类型

## 麻将

地方特色的游戏，本章介绍具体的规则；高县麻将采用宜宾地区流行的麻将打法，在宜宾及周边比较盛行。核心是打缺门，翻精（一头精和两头精）,血战模式，高县计番，流局查叫和查花猪。

### 游戏规则

1. 只有条（索）、筒（饼）、万三种牌共108张，没有花、风牌和箭牌。(ok)
2. 不可以吃。
3. 动作面板上"过"表示放弃，放弃碰、放弃贴等操作(ok)
4. 必须缺门可胡，即胡牌的时候不能有三种花色的牌。(ok)
5. 最后四张自动胡，即当牌墙只剩下最后4张的时候，如果玩家的手牌已经下叫，而叫牌正好是这4张中的任意一张，那么当这张牌出现时，叫的玩家只能胡牌，而不能跳过胡牌去进入后面的查叫过程，玩家没有选择的权利。(ok)
6. 一炮多响。即某个玩家打出一张牌，有不止一个的玩家要之成胡，则均可胡牌。(ok)
7. 引杠：就是玩家手中有三张一样的牌，当其他玩家打出了第四张同样的牌时，造成持三张同样牌的玩家杠上花而胡牌，打出第四张同样牌的玩家属于引杠，由该玩家一人承担所扣游戏币。（注：该项虽与高县麻将不同，如此设置仅为避免作弊。）
8. 自杠（包括暗杠），就是玩家已经碰了三张一样的牌时，玩家自己又摸到了第四张一样的牌，这时选择的杠牌。或者玩家手里有四张同样的牌而选择杠牌，如杠上花，则所得游戏币由另三家平均承担。

手上有碰的牌或者贴的牌，不能用听用来杠。

如果手上有贴的牌，不能用听用来杠

1. 抢杠。就是玩家已经碰了三张一样的牌，自己又摸到了第四张一样的牌，玩家选择下杠时，如其他玩家可以胡这张牌，就可以抢杠而胡，胡牌基数增加1倍，由杠牌玩家付给游戏币。注：暗杠不能抢杠胡牌。
2. 流局查花猪查大叫普通场中，流局指没人胡牌。血战场中，流局指最后还有两个人或者以上没有胡牌。

流局处理：

* 第一步：查卫生（花猪）

手上拿着3门牌的玩家为花猪，花猪赔给非花猪玩家8倍基数游戏币。不管非花猪有没有叫是否胡牌，只要不是花猪就可以得到赔偿，？

* 第二步：查大叫

没听牌的玩家（花猪不用）赔给听牌的玩家最大的可能番（大叫）。

查叫： 逃跑玩家不赔偿，胡牌玩家不赔偿，对自己不赔偿

1. 血战到底：

血战场中的血战模式指1家胡了并不结束该局，而是未胡的玩家继续打，直到有3家都胡或者余下的玩家家流局。这样先胡的玩家家不一定获利最多，点炮的玩家也能翻身，提高了博弈性和趣味性，促进了牌局的稳定发展。牌局结束，一并结算，如果胡了的玩家提前离开，则不给予所赢番数。

1. 坐庄

* 开盘第一局由电脑随机选择玩家投骰。 只要有玩家变动视为第一局。
* 非开盘第一局由前一盘第一个胡牌的玩家投骰，如第一胡牌玩家有两个以上（一炮多响），由放炮玩家投骰。

1. 精（翻听用）

* 单精：每盘所有玩家发完牌后，桌上余牌的第一张翻开，该数字同花色即为精，该数字依次后面一个同花色的为听用。举例：如翻4筒，即4筒为精，5筒就是听用。如遇9则顺延。如翻9万，则9万是精，1万就是听用。
* 双精：每盘所有玩家发完牌后，桌上余牌的第一张翻开，该数字同花色即为精，该数字依次前面和后面一个同花色的为听用。举例：如翻4筒，即4筒为精，3筒和5筒就是听用。如遇1或9则顺延。如翻9万，则9万是精，8万和1万就是听用；如翻1万，则1万是精，2万和9万就是听用。

1. 听用

可以代替任意一张牌。打出听用牌，进入托管状态且不能胡牌，牌局结束后，不差花猪，但是要查叫，如果所有玩家都进入打出听用状态，那么是否需要提前结束游戏，按照查叫规则来查询（这个暂时还未实现）。

听用只能用来：胡，贴，提，不能用来杠。

1. 动作状态

碰

贴

提

杠

胡

用户操作牌类型如下

EstimatKind\_OutCard, //出牌效验

EstimatKind\_PengGangCard, //杠牌效验

EstimatKind\_TICard, //提牌校验

EstimatKind\_TIECard, //贴牌校验

EstimatKind\_PENGCard, //碰牌校验

1. EstimatKind\_OutCard 当前用户出牌

出牌后，所有用户可能的动作

碰牌、贴牌、杠牌、胡牌

1. EstimatKind\_PENGCard当前用户碰牌

碰牌后，估计当前用户的动作

提、杠

1. EstimatKind\_TIECard当前用户碰牌

碰牌后，估计当前用户的动作

提、杠

1. EstimatKind\_TICard 当前用户提牌

提牌后，估计当前用户动作

杠、胡

1. EstimatKind\_PengGangCard 当前用户杠牌

杠牌后，估计当前用户动作

提、胡

动作响应等级

估计用户响应后，根据可能的用户响应，这里有能用户有相同的处理动作，对不同等级不需要再区分，对相同等级，需要做出区分。

1. 同时贴,有不同的优先级，离出牌用户越近等级越高。
2. 同时吃胡，允许同时吃胡，不需要判断优先级。
3. 抢杠（吃胡与杠同时出现，一家杠，一家或者两家吃胡，杠胡等级不同）
4. 贴胡同时出现，但是贴胡等级不同。

综合各种情况，只有同时贴牌才需要区分离出牌用户的位置

### 计分规则

* 每有一杠加一番；
* 不含牌精加一番；
* 有三个牌精加一番；
* 打出听用，不能和牌；
* 查花猪，包其他玩家极品；
* 查叫，没有落叫的玩家，需要对已经落叫的玩家进行最大牌面的赔偿；
* 查花猪、查叫，已经胡牌的玩家不在赔偿范围

## 斗地主

### 游戏流程规则

### 计分规则

# 开发事项

## 麻将

### 庄家设置

在有用户离开后，新用户尽量，随机产生一个庄家，如果没有用户离开，开始新局，设置上局赢家为当前庄家。也就是用户离开，需要设置当前庄家不确定。

开局

随机庄家

继续游戏

用户离开

用户进入

赢家庄家

### 游戏结束

游戏介绍有三种情况：正常结束、强制解散、用户强退

#### 正常结束

正常结束，需要统计分数，包括逃跑、杠分，查花猪，查大叫等，下面描述其流程：

* 构造CMD\_S\_GameEnd结构类型信息，后面会发送该结构数据到客户端进行处理。
* 流局的计算：

判断当前玩家数量，判断吃胡权，没有吃胡权的玩家数量大于一，说明是流局

不管是否有流局，依次对未胡牌的在线玩家（吃胡权为空）进行查叫查花猪，其他的玩家包括（逃跑玩家，已经胡牌玩家【吃胡权不为空】）。

1. 判断是否在游戏中

已经胡牌用户或者逃跑用户都不在游戏中，只对游戏中的用户进行查花猪、查大叫处理。

1. 查花猪

游戏结束的时候，只能包含两种花色，否则为花烛，赔偿规则：

1. 查大叫

游戏结束后，不能胡牌，但不是花猪，没有下叫的用户，赔偿规则：

* 统计积分

赔偿分数、牌面分数

* 发送游戏结束命令

SUB\_S\_GAME\_END 附带CMD\_S\_GameEnd结构类型数据到客户端所有用户。

### 游戏逻辑

#### 服务器端

CTableFrameSink::OnUserOperateCard 用户出牌响应

1. 检查输入是否合法，不合法直接返回
2. 对当前用户m\_wCurrentUser分为两类，主动用户和被动用户。

wChairID 是操作牌的用户。

1. 主动用户

if (m\_wCurrentUser==wChairID)

如果是用户摸了一个牌之后的操作，比如杠等，主动权在操作牌用户的手上，比如杠牌之后，可以胡，可以提等操作。

第一步： 检查操作码cbOperateCode，操作牌cbOperateCard的有效性。检查用户操作的有效性（打出听用的用户不能继续操作，虽然客户端已经限制，但是还需要在服务器端做处理）

第二步： 根据操作码cbOperateCode的类型做switch 分支流程处理

case WIK\_TI\_PAI: //提牌操作

case WIK\_GANG: //杠牌操作

case WIK\_CHI\_HU: //吃胡操作

第三步 ：结束

1. 被动用户

if (m\_wCurrentUser==INVALID\_CHAIR)

如果其中一个用户打出了牌，当前用户设置为无效，因为还不知道谁会响应当前打出的牌，这种情况是被动的操作。因此被动是指对其他用户的响应。

被动处理的过程中，会出现，对请求用户的等级的比较，以便于判断谁最终操作有效

第一步：检查当前请求的用户是否已经响应过，检验用户传递过来的操作码是否与服务器上保留的预估的操作码之一，如果条件不符合直接返回，不进行任何处理。

第二步：WORD wTargetUser=wChairID; 把当前请求操作的用户设置为目标用户wTargetUser 。 BYTE cbTargetAction=cbOperateCode;把当前请求操作的用户的操作码设置为目标操作码。主要是为了做操作等级判断。

m\_bResponse[wChairID]=true; 设置当前用户wChairID已经被响应过。

m\_cbPerformAction[wChairID]=cbOperateCode; 记录下当前用户从可操作码列表中选取的一个操作码，比如可以碰杠贴，用户只选择了杠，那么实际执行的操作码是杠，它就被记录下来。m\_cbOperateCard[wChairID]记录下操作牌数据。

判断在自己未摸下一张牌的一圈内,不能弃上一家而胡下一家。

判断其他用户的操作动作

for (WORD i=0;i<m\_wPlayerCount;i++) // 检查所有用户

{

设置用户动作 cbUserAction：

检查第i个用户，如果已经响应过，设置用户动作cbUserAction为用户选择的操作码，如果没有响应过，设置用户动作为可能的操作码列表。

取得第i个用户动作等级 cbUserActionRank：

取得当前操作用户的动作等级cbTargetActionRank

//开始进行等级判断，需要对可能相同等级的进行判断

第一个，贴牌WIK\_TIE\_PAI

如果当前请求用户是贴牌操作cbTargetAction & WIK\_TIE\_PAI，比较第i个用户的动作

依据出牌用户m\_wProvideUser，更具出牌用户的位置依次计算所有用户的等级，离出牌用户最近的用户，等级越高，依次递减。

如果当前请求用户的等级低于第i个用户操作的等级

wTargetUser=i;cbTargetAction=cbUserAction; 把高优先级用户及其动作，设置为目标可响应动作的用户。如果目标用户wTargetUser 还没有请求响应，那么就先返回，等待没有响应的用户进行响应。

}

第三步，通过比较用户操作的优先级，目标用户已经产生后，需要执行最终用户的动作。

检查目标用户cbTargetAction 是吃胡操作类型，并且其中还有用户可以吃胡但是还未响应的用户，等待可以吃胡的用户响应

如果目标用户是放弃操作，cbTargetAction == WIK\_NULL，从所有用户返回的结果来看，没有人进行任何操作，因此。发牌给下一个用户并返回。

取得目标用户的操作牌cbTargetCard

胡牌动作：

循环检查用户是否吃胡动作，如果是吃胡动作，处理吃胡用户。cbTargetAction 目标动作类型是吃胡，需要判断在该类型下每个用户的动作m\_cbPerformAction[i]是否是吃胡，如果是吃胡类型，处理完吃胡后，判断是游戏是否结束，如果没有结束发牌给下一个用户。（抢杠，抢提失败的原因？）

开始

检查下一个玩家

游戏中

跳过

花猪

查叫

是

是

花猪处理

无叫处理

无

打出听用

是

分数处理：

对用户全部积分不够赔偿其他玩家的情况下，仅仅只需要支付现有的积分。送分是在每局的ready 后开始送分，如果分数不够最低分数，ready前需要判断还有没有送分次数。

送分：

考虑到已经胡牌的玩家不参与到赔偿范围，那么不管是怎么样的情况下，用户都只可能赔偿三家极品的情况：

如果分数不够赔偿，将玩家手上的剩余分数平均赔偿给其他玩家（金币积分都相同）。

取得当前用户

花猪用户

赔偿其他非花猪用户

是

无叫用户

剩余分数不够

不够分结算

是

赔偿其他有叫用户

剩余分数不够

是

正常结算

结束

否

否

#### 强制解散

解散游戏是超级管理员在聊天语句中发送结束停止命令后执行，简单初始化桌面数据

#### 用户强退

当其中的一个用户逃跑的时候，会触发到该部分，下面看看其触发的流程

当客户端发起离开桌子的命令后，服务器端CTableFrame::PerformStandUpAction将响应用户的请求，在其中判断if (IsUserPlaying(pIServerUserItem)==true)条件，如果满足的话，那么就说明是正在玩的过程中逃跑。m\_pITableFrameSink->OnEventGameEnd(wChairID,pIServerUserItem,GER\_USER\_LEFT);

表示用户逃跑，但是当前的游戏在用户逃跑后仍然要继续下去，因此该调用并不真正的结束游戏，而是扣除逃跑玩家的分数后继续游戏。通知其他用户逃跑情况。

如果用户逃跑，赔偿其他三个玩家满分后，那么就要去玩家开局前至少要有赔偿三个满分的分数，这个是开局的要求。

是否需要单独记录逃跑赔偿得分？