**吹牛模式**

|  |  |  |  |
| --- | --- | --- | --- |
| **创建人名** | **修改人名** | **时间** | **修改内容** |
| 杨靖 |  | 2021-10-22 | 创建文档 |
|  |  |  |  |
|  |  |  |  |

# 界面相关

## 吹牛相关

* 同已有界面

## 竞猜相关

* 不要用弹板，操作直接在【牌桌】上实现

# 逻辑相关

## 游戏开始

### 创建新游戏

* 房间内的所有玩家都加入该游戏
* 具体逻辑在【房间&匹配】已描述

### ~~扣除底注~~

* ~~每个玩家都要从自身金币中，扣除该底注金额~~
* ~~根据房间编号，读RoomConfig【房间配置】表的【底注金币】字段~~

## 摇骰子

### 系统摇骰

* 给每个参与者摇骰子
  + 3个6面骰

### 玩家重摇

* 在第一次玩家叫点之前，都可以重摇
* 重摇需要额外扣除【底注金币\*本次重摇消耗比例】，并均分给其他玩家
  + ReRollCostConfig【重摇消耗配置】表的【吹牛模式消耗金币比例】
* 重摇次数越多，消耗越大
  + 见ReRollCostConfig【重摇消耗配置】表

## 游戏过程

### 基本流程

* 玩家按照座位顺序，轮流叫点
  + 直到最大点数或是有人开/抢开为止

### 回合时长

* 读RoomConfig【房间配置】的【回合时长】字段

### 叫点

* 叫点简介
  + 叫点由【点数】+【数量】构成
* 叫点规则
  + 最小玩家可以喊的【数量】=玩家人数
  + 最大玩家可以喊的【数量】=玩家人数\*6（每一个玩家摇到豹子）。
  + 在一个有7名玩家的房间中，这些数字是: 7-42
* 若回合内玩家未叫点
  + 超时后，如果在本回合内，玩家在前端有手动操作，则后端以该叫点替玩家叫点
    - 意味着前端每次变更【数量&点数】，都需要向后端发消息并记录
  + 否则，后端自动替玩家叫点
    - 开局后，后端初始默认桌面点数为2，数量为参与者数量
    - 若当前数量未到上限，则点数继承桌面上的点数，数量比当前桌面上的+1
    - 如果数量已经到上限，则数量与桌面上的相同，点数比当前桌面上的点数+1
* 若最大点数已经被叫过
  + 则系统自动替下一个无点可叫的玩家开
* 游戏过程中，前端的点数&数量会根据当前叫点情况实时变化
  + 进入房间后，前端初始默认点数为2，初始数量为参与者数量
  + 若当前数量>桌面上的数量，则数量、点数不变，否则
    - 点数不变，数量比桌面上的数量+1
  + 如果桌面上的数量已经到上限，则数量与桌面上的相同，比当前桌面上的点数+1

### 开/抢开

* 开/抢开倍率取决于跳过的玩家
  + 正常【开】视为跳过0个玩家
  + 读SystemConfig【系统常量配置】的【开/抢开倍率】
* 抢开、被抢开的玩家，若其拥有的金币，不能够支付抢开带来的额外下注差值
  + 则以双方所拥有的【最小金币数】，作为本次【开/抢开】对决所用的金额
  + 否则以【底注\*本次开/抢开倍数】，作为本次【开/抢开】对决所用的金额
* 开/抢开方与被开方，都锁定该金额的金币，直至吹牛结算结束
* 开/抢开后进入竞猜环节

## 游戏规则参考

### 点数大小

* 点数强度：1>6>5>4>3>2
* 在实际游戏场景中这意味着：
  + 一个玩家喊7个1，我没有其他7个点数可以喊，只能喊8个2以上
  + 相似，如果有玩家喊7个2，我可以喊7个3/4/5/1

### 结算时的1点

* 在结算时，1代表被开的点数
  + 比如：玩家喊10个5被开了，桌子上所有的1会变成5，除非1已经被叫过了

### 豹子/顺子

* 玩家摇到豹子/顺子的时候开牌的时候播放豹子/顺子动画
  + 动画不应该影响游戏的节奏
  + 豹子：该点数数量+1
  + 顺子：不算任何点数

## 竞猜环节

### 竞猜简介

* 房间内的所有玩家【参与者】、旁观者都可以押注竞猜胜败
  + 但别人压了什么是黑箱不可见，只能看见总押注

### 竞猜底注

* 竞猜底注读RoomConfig【房间配置】的【底注金币】字段

### 押注倍率

* 旁观者与参与者的押注倍率，均为本局游戏的对决倍率
  + 例如本局抢开了16倍，则所有人押注，均以16倍作为押注倍率

### 押注规则

* 每次押注前需判断玩家当前拥有金币数
  + 如果小于【底注金币\*押注倍率】，则不可押注
* 每次押注成功后，扣除对应的金币数
* 允许玩家多次下注
* 允许进行对冲下注【同时压胜/败】
* 允许对自己下注

### 竞猜时间

* 竞猜时长读RoomConfig【房间配置】的【竞猜时长】字段
* 到达时间后，竞猜截止，进入结算环节

## 结算&结束

* 竞猜结束后，开骰盅，并结算

### 吹牛结算

* 根据规则，判断谁胜利
  + 若叫点是否大于场面点数，开/抢开者胜利
  + 否则被开方胜利
* 根据结果，胜利者获得失败方的对应倍率筹码
  + 抢开部分的筹码输赢逻辑见【1.3.4 开/抢开】
* 其他玩家的筹码退还

### 竞猜结算

* 在吹牛结果展示前，根据吹牛的结果，确定竞猜的输赢
* 竞猜胜利方，可以瓜分失败方的总注池
  + 根据押注多少，按比例瓜分失败方的注池
  + 如胜利方总共100倍，A压了20倍，则A拿走20%的失败方注池

### 盈利抽税

* 只对玩家盈利的钱进行抽税
  + 抽税=Int(盈利\*TaxRatio%)
* 吹牛和竞猜的盈亏，分开抽税
  + 竞猜内每笔单独抽税，比如对冲押注，押中部分仍然会被征税

### 离线通过邮件发送收益&损失通知

* 若玩家在结算前离线，则通过邮件发送盈利&损失通知
  + 对局的盈利，使用编号为【1】的邮件模板，盈利金额传导至邮件正文的动态参数，盈利金币放入附件
  + 对局的损失，使用编号为【2】的邮件模板，损失金额传导至邮件正文的动态参数
  + 竞猜的盈利，使用编号为【3】的邮件模板，盈利金额传导至邮件正文的动态参数，盈利金币放入附件
  + 竞猜的损失，使用编号为【4】的邮件模板，损失金额传导至邮件正文的动态参数
  + 没有对手导致竞猜流产，使用编号为【5】的邮件模板，返还金额传导至邮件正文的动态参数

### 结算动画

* 开/被开的玩家，按照吹牛的结果显示输赢
* 其他人按照竞猜的结果显示输赢

### 游戏结束

* 播放结算动画后，游戏结束

## 其他行为

### 升场

* 玩家在游戏中可以发起【升场】
  + 发起&同意前，都需要检测玩家自身金币是否大于等于该底注金额
  + 如果所有本场游戏的【游戏中】玩家都同意
    - 那么下一场的RoomLv+1，RoomId变更
    - 不同意的【等待游戏】玩家，会被强制站起
  + 否则，提案失败
* 处于升场决策过程中的房间，会被匹配系统临时屏蔽
* 升场逻辑加入限时规则
  + SystemConfig【系统常量配置】新增字段【升场限时】，超时后自动否决
  + 如果是发起投票后再坐下的玩家，超时默认同意
    - ~~另，下局开始前，也自动否决~~
  + ~~前端的否决/确认状态，在下局开始后，清除，飘字逻辑不受影响~~
* ~~升场逻辑限次规则~~
  + ~~一局【从对局开始——下一局开始前（包含对局间隔）】中，全局只允许发起一次升场。~~
  + ~~发起后，再次点击升场按钮，返回提示【本局已发起过升场】~~

### 退出游戏

* 玩家在游戏过程中，可以申请退出游戏
  + 玩家会在游戏结束后，退出游戏
    - 玩家可以取消该申请
  + 玩家胜利/失败，都会获得对应的奖励/损失
* 退出游戏的同时也会退出房间

## 动画&提示

### 匹配中

### 对局开始

### 轮到你了

### 开

### 抢开

### 胜利

对决者【2人】中胜利方播放

### 获利

非对决者，最终收益为正，播放

### 失败

对决者【2人】中失败方播放

### 损失

非对决者，最终收益为负，播放

## 流程

### 游戏开始&摇骰-2秒

* 前端并发播放：【游戏开始】动画、【摇骰盅】动画
* 后端执行待时：SystemConfig【系统常量配置】的【RollDiceWaitTime】
* 之后进入回合轮转阶段

### 回合-10秒

* 前端并发播放：【我的回合】动画，同时开始倒计时【RoundLength】
  + 如果是首个回合内，可以一直重摇
* 后端执行待时：RoomConfig【房间配置】的【RoundLength】
* 开/抢开后，进入【开/抢开】阶段

### 开/抢开-2秒

* 前端播放：【开/抢开】动画
  + 抢开者本人，需要一个额外的动画将抢开倍数驻留在自己身边
* 后端执行待时：SystemConfig【系统常量配置】的【OpenDiceWaitTime】
* 之后进入【竞猜】阶段

### 竞猜-5秒

* 前端播放：【竞猜】界面，同时开始倒计时【VoteLength】
* 后端执行待时：RoomConfig【房间配置】的【VoteLength】
* 之后进入【结算】阶段

### 结算-7秒/5秒

* 前端播放：【拉起他人骰盅&播放胜利/失败/盈利/损失动画】，有效骰子高亮，展示总骰子数
* 后端执行待时：SystemConfig【系统常量配置】的【FinishGameWaitTime】
* 之后本局游戏结束

### 局间间隔-3秒

* 前端播放：清上一局的骰子，并且恢复骰盅状态
* 后端执行待时：RoomConfig【房间配置】的【GameIntervalTime】

# 配置相关

## RoomConfig【房间配置】

## ReRollCostConfig【重摇消耗配置】

## SystemConfig【系统常量配置】