

如何快速开发小游戏之 排行榜

By 毅然

目录

- 上期回顾
- 本期目标
- 游戏逻辑
- 代码开发

上期回顾

<http://answer-game.leanapp.cn/>

上期回顾

- SDK 安装
- 云引擎 - 云函数

本期目标

[> 返回](#)

世界排名

| 排名 | 用户名 | 分数 |
|----|-----|-----|
| 0 | 常秀英 | 780 |
| 1 | 邱刚 | 710 |
| 2 | 田霞 | 650 |
| 3 | 易军 | 630 |
| 4 | 龚勇 | 620 |
| 5 | 苏强 | 600 |
| 6 | 阎刚 | 570 |
| 7 | 吴丽 | 560 |
| 8 | 易丽 | 530 |
| 9 | 方丽 | 470 |

您的排名

| 排名 | 用户名 | 分数 |
|----|-----|-----|
| 12 | 唐杰 | 360 |
| 13 | 孟桂英 | 300 |
| 14 | 萧勇 | 200 |

游戏逻辑

- 分数计算
 - 单人模式每天最多答题 3 次。
 - 每答题一次，累计计算成绩，将答题总分计入排行榜中。
 - 答题总分越高，排名越高。
 - 每周数据重置，重新开始答题排名。
- 排行榜界面展示
 - 世界 top 10 玩家
 - 当前玩家及附近的排名
- 发放奖品：
 - 每周根据上周结果发放奖品，前 10 名每人获得 10 金币。

代码开发

代码开发 — 服务端

限制答题次数

```
var countQuery = new AV.Query('ChallengeQuestion');
var today = new Date((new Date()).setHours(0, 0, 0, 0));
countQuery.equalTo('creator', request.currentUser);
countQuery.greaterThanOrEqualTo('createdAt', today);
return countQuery.count().then(function(count) {
    if (count >= 3) {
        // 玩过 3 次就不让玩了
        console.log('抛出错误');
        throw new AV.Cloud.Error('Exceeded Limit', {code: 403});
    } else {
        // 选题逻辑
        .....
    }
});
```

创建排行榜

- 名称: world
- 重置周期: week
- 排序: descending
- 更新策略: sum

代码开发 — 客户端

更新分数

```
// 更新分数前需要用户登录
var score = 0;
for( var i = 0; i < this.userOptionScores.length; i++ )
{
    score += this.userOptionScores[i];
}

AV.Leaderboard.updateStatistics(AV.User.current(), {
    world: score,
});
```

获取排名

世界 Top 10

```
var leaderboard = AV.Leaderboard.createWithoutData('world');  
return leaderboard.getResults({  
  limit: 10,  
  selectUserKeys: ['username'],  
});
```

获取排名

当前玩家附近的排名

```
var leaderboard = AV.Leaderboard.createWithoutData('world');  
return leaderboard.getResultsAroundUser({  
  limit: 3,  
  selectUserKeys: ['username'],  
});
```

发放奖品

需求

- 每周一凌晨发放奖品
- 前 10 名用户每人可以获得 10 金币

实现方式

- 云引擎定时任务

发放奖品

```
AV.Cloud.define('weeklyRewards', function(request) {
  var leaderboard = AV.Leaderboard.createWithoutData('world');
  // 查询当前排行榜的版本
  return leaderboard.fetch()
    .then((leaderboard) => {
      // 当前版本减去 1 是最新的历史存档数据
      var lastHistoryVersion = leaderboard.version - 1;
      // 获得上一版排行榜中的前 10 名用户
      return leaderboard.getResults({
        limit: 10,
        version: lastHistoryVersion,
      });
    })
    .then((rankings) => {
      // 给每个用户增加 10 个金币。
      var users = rankings.map((ranking) => {
        var user = ranking.user;
        user.increment('gold', 10);
        return user;
      });
      return AV.User.saveAll(users);
    })
    .....
  });
```

发放奖品

查看效果

- 手动重置排行榜
- 运行代码

预告

多人实时在线对战