

REPORT

基于智能合约的分布式大富翁游戏

June 18, 2020

姓名: 杨晨宇、蓝宇霆

学号: 517030910386、517030910368

Contents

1	项目介绍	2
1.1	背景	2
1.2	相关工作	2
2	项目环境	2
3	项目实施	2
3.1	合约实现	2
3.1.1	数据结构	2
3.1.2	合约事件	2
3.1.3	主要函数	2
3.2	前端交互	3
4	项目难点	3
5	后续工作	3
6	分工细节	3
7	参考资料	3
8	完整代码	3

1 项目介绍

1.1 背景

1.2 相关工作

在实现该项目的过程中，我们主要参考了Truffle官方文档[1]和Web3.js官方文档[2]。此外，由于编写合约的Solidity及与合约的交互过程对我们而言都较为陌生，因此我们也阅读了许多开源项目的源代码，其中包括官方样例Pet-Shop[3]以及非常有名的入门项目僵尸工厂[4]等等。虽然这些项目都与我们要做的大富翁有一定区别，但通过它们我们对于智能合约的应用有了更深刻的了解。

此外，我们还参考了github上一个利用智能合约实现五子棋的项目[5]。该项目与我们的目的较为接近——事实上，我们的房间创建/加入的想法就脱胎于此，并在其基础上从双人扩展到多人。但该项目版本过于古老，且游戏机制远不如大富翁复杂，因此也只是作为参考，并未用于实际实现中。

我们的主体框架是在官方样例webpack-box[6]的基础上搭建的。该项目原本仅用来展示虚拟货币的支付过程。我们采用了它的目录结构，而其他部分，包括前端设计，页面交互，事件广播及合约逻辑最终全都由我们自己实现。

2 项目环境

3 项目实施

3.1 合约实现

3.1.1 数据结构

3.1.2 合约事件

3.1.3 主要函数

```
1 function createRoom(uint32 _roomId) public payable ;
```

3.2 前端交互

。。。。在创建房间/加入房间以后，根据当前房间人数，玩家会被分配一个座位号。考虑到实际测试是在私链上进行，我们利用房间号+座位号的方式唯一确定一名玩家的身份，并利用他的地址作为验证。

4 项目难点

5 后续工作

6 分工细节

7 参考资料

1. Truffle官网 <https://www.trufflesuite.com/docs>
2. Web3.js官网 <https://web3js.readthedocs.io/en/v1.2.9/>
3. 官方样例Pet-Shop <https://github.com/truffle-box/pet-shop-box>
4. 僵尸工厂 <https://cryptozombies.io/en/course>
5. 利用智能合约实现五子棋对战 <https://github.com/Alexygui/Gobang>
6. WebPack <https://github.com/truffle-box/webpack-box>

8 完整代码

<https://github.com/Achronferry/SJTU-EE357-project-BlockChain>