

# 印加宝藏区块链游戏项目介绍文档

<https://github.com/Auspelan/IncanGold>

## 1. 项目概述

印加宝藏是一款基于区块链技术的多人在线冒险游戏，玩家扮演探险家在神秘的印加神庙中寻找宝藏。游戏融合了策略、风险管理以及区块链技术，通过智能合约确保游戏过程的透明性和公平性。玩家需要在每轮游戏中决定是继续探索获取更多宝藏还是安全返回营地，同时要警惕路上的陷阱。游戏最终通过区块链智能合约进行结算，确保奖励分配的公正透明。

## 2. 游戏规则

游戏需要在线匹配3个玩家，需要投入一定入场费才能入场。

进入游戏后每个玩家初始在营地拥有0个金币。

游戏一共进行3轮。

每轮游戏按照下列阶段的顺序进行：

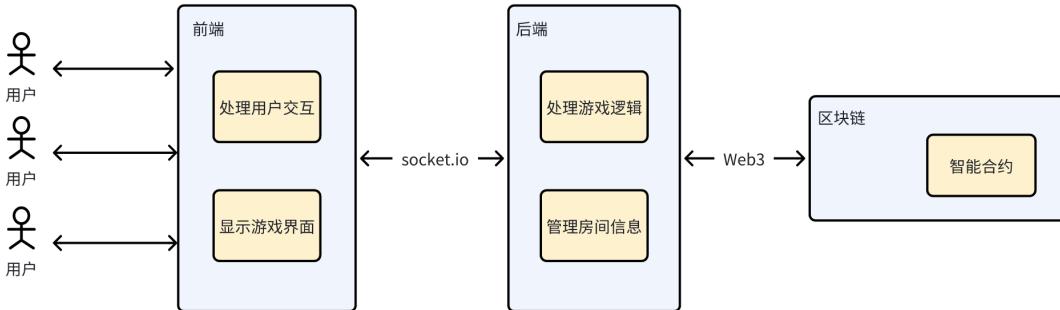
- 选择阶段
  - 玩家选择走一步或者回营地
  - 玩家无法看到其他玩家的行为
  - 所有玩家选择完毕后进入判定阶段
- 判定阶段
  - 若玩家选择返回营地，可以获得沿路所有地块上剩余的金币。若多名玩家同时返回则多名玩家需平分金币
  - 判定是否还有玩家在路上，若无玩家在路上进入结算阶段
  - 判定下一地块是陷阱或者奖励
    - 若是陷阱则遭遇陷阱数（每轮游戏开始时为0）加1
      - 若遭遇陷阱数为2则所有尚未返回营地的玩家丢失携带的所有金币，并强制返回营地。进入结算阶段
      - 若是奖励则将当前地块上的金币平均分给所有还在路上的玩家，未能整除的部分留在当前地块。进入选择阶段
- 结算阶段
  - 将玩家身上的金币转移到营地
  - 判断是否已经经过3轮游戏，若已经过3轮游戏则游戏结束，进行最终结算
    - 三名玩家投入的入场费在扣除gas费用后按照他们在游戏中获得的总金币数进行同比例分配
  - 进入新一轮游戏的选择阶段

## 3. 技术开发方案

## 3.1 系统架构

我们的印加宝藏游戏采用三层架构：

1. **前端层**: 使用Vue.js框架构建响应式用户界面，提供游戏交互和状态展示。
2. **后端层**: 使用Node.js和Express框架处理游戏逻辑、用户匹配和实时通信。
3. **区块链层**: 使用以太坊智能合约处理游戏结算和资产转移。



## 3.2 技术栈

### 前端技术

- **框架**: Vue 3 + Pinia 状态管理
- **样式**: CSS3 + 玻璃拟态设计
- **通信**: [Socket.io](#) 客户端
- **构建工具**: Vite

### 后端技术

- **运行时**: Node.js + Express
- **实时通信**: [Socket.io](#)

### 区块链技术

- **智能合约**: Solidity
- **开发框架**: Truffle
- **测试网络**: Ganache本地开发网络

## 3.3 核心技术实现

### 3.3.1 智能合约设计

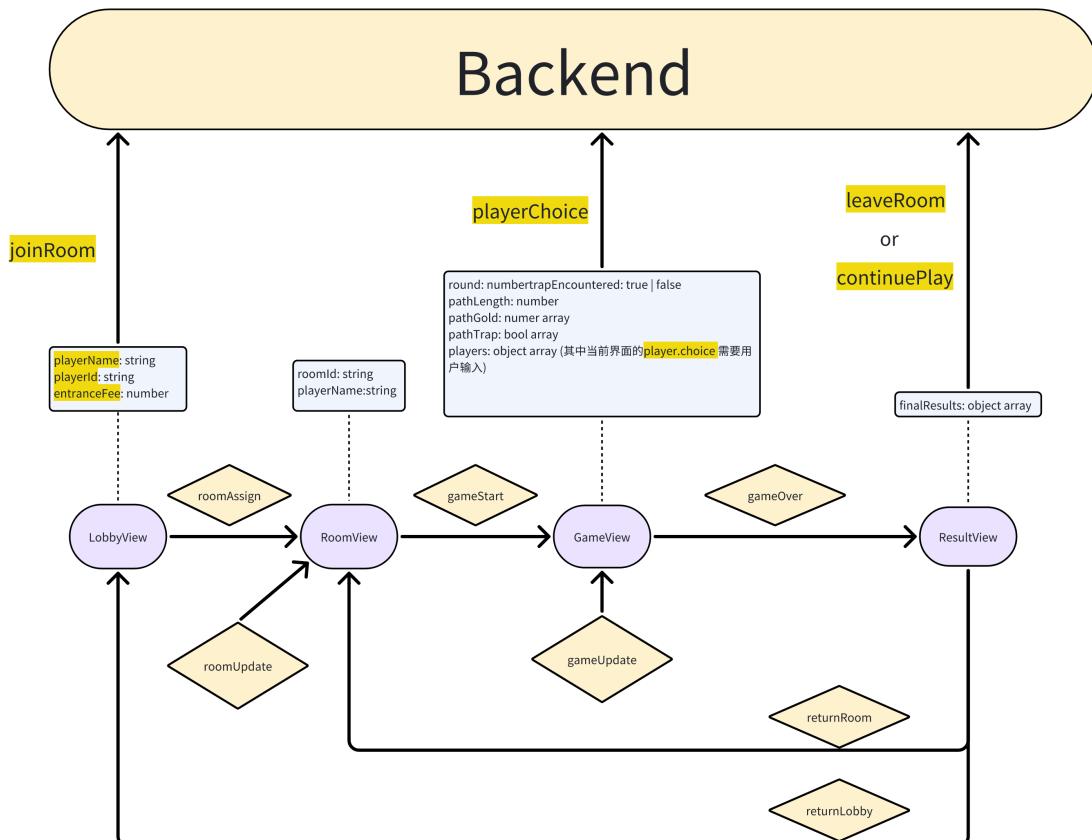
我们设计了 `IncanGold.sol` 智能合约，它作为整个游戏经济系统的中心，负责以去中心化、透明且安全的方式管理所有游戏资金。

- **核心数据结构**: 使用 `Game` 结构体封装单场游戏状态，并通过 `mapping` 支持多游戏并发。
- **资金管理**: `joinGame` 函数确保玩家支付等值入场费，资金进入合约控制的奖池。
- **自动结算**: `settleGame` 函数由后端（owner）调用，根据计算结果自动将奖励分配给玩家。
- **透明性与可审计性**: 所有关键操作通过事件记录在链上，完全公开可追溯。

### 3.3.2 前后端交互

为了实现多人游戏的实时性，前后端之间采用**Socket.IO**技术进行双向、实时的通信。这种架构确保了：

- **游戏状态同步**: 当任何玩家做出决策后，后端会立即将最新的游戏状态（如玩家位置、金币变化、当前回合等）广播给房间内的所有玩家，保证所有客户端看到的信息完全一致。
- **玩家动作广播**: 玩家的所有操作（如加入房间、做出选择）都会通过Socket事件实时发送到后端进行处理，避免了传统HTTP轮询的延迟。
- **游戏流程驱动**: 游戏的关键节点，如游戏开始、回合结束、游戏结算等，都由后端通过特定事件通知前端，驱动前端界面的切换和更新。



### 3.3.3 后端与区块链交互

后端服务器作为连接游戏逻辑与区块链的桥梁，通过**Web3.js**库与部署在以太坊网络上的 `incanGold` 智能合约进行交互。我们创建了一个专门的 `blockchain.service.js` 服务来封装所有链上操作，其主要职责包括：

- **玩家入金处理**: 当玩家确认加入游戏时，后端服务会调用智能合约的 `joinGame` 方法，将玩家的入场费安全地转入合约地址，完成资金的锁定。
- **游戏结算执行**: 当一局游戏结束，后端根据游戏逻辑计算出最终排名和奖励分配方案后，会调用合约的 `settleGame` 方法。该方法将根据后端提供的获胜者地址和奖励金额，自动、透明地将合约中的资金分配给相应玩家。
- **动态Gas管理**: 为了确保交易能被及时处理，后端在每次发起链上交易前，都会动态获取当前网络的Gas价格，优化了交易的成功率和成本。
- **信息查询**: 后端通过调用合约的视图函数（如 `entryFee`）来获取链上配置，确保前后端信息的一致性。

通过这种设计，我们将复杂的区块链操作封装在后端，为前端提供了简洁的API接口，同时保证了链上操作的安全性和可靠性。

## 4. 最终成果

### 4.1 功能实现

基于项目代码分析，我们成功实现了以下核心功能：

#### 1. 智能匹配大厅系统：

- 玩家可通过 `LobbyView.vue` 输入昵称和钱包地址，选择不同档位的入场费（0.01/0.1/1 ETH）
- 后端 `Manager.js` 自动进行玩家匹配，优先填充已有房间，满3人自动开始游戏
- 实时显示房间状态和已加入玩家列表，提供清晰的匹配进度反馈

#### 2. 完整的多人游戏流程：

- 房间等待阶段** (`RoomView.vue`)：显示所有玩家准备状态，支持玩家离开房间
- 游戏进行阶段** (`GameView.vue`)：实现多轮探险机制，每轮包含：
  - 玩家同时选择前进或返回 (`PlayerActions.vue`)
  - 实时游戏面板显示 (`GameBoard.vue`)，包括路径金币、陷阱位置和玩家分布
  - 自动状态同步和事件提示系统
- 结果结算阶段** (`ResultView.vue`)：显示最终排名和链上收益分配

#### 3. 区块链资金托管与结算：

- 玩家加入游戏时通过 `joinGameOnChain` 将入场费锁定至智能合约
- 游戏结束后基于金币数量比例计算收益，通过 `solveGameOnChain` 自动分配奖池
- 所有资金流转在链上完全透明可追溯

#### 4. 实时状态同步系统：

- 基于 `Socket.IO` 的 `roomUpdate`、`gameUpdate`、`gameOver` 等事件实现毫秒级状态同步
- 前端 Pinia store 的 `applyGameSnapshot` 确保多客户端状态一致性
- 连接异常时的自动重连和状态恢复机制

### 4.2 技术成果

#### 1. 安全可靠的智能合约：

- `IncanGold.sol` 合约通过完整的Truffle测试套件验证，涵盖权限控制、资金安全和边界情况
- 实现仅Owner可触发的安全结算机制，防止恶意操作
- 优化的Gas消耗设计，降低玩家交易成本

#### 2. 高性能游戏服务器：

- 基于 `Manager.js` 的游戏会话管理，支持多房间并发运行
- `game.js` 游戏引擎实现完整的回合制逻辑，包括陷阱触发、金币分配和回合推进
- 专门的 `blockchain.service.js` 服务层封装所有Web3交互，提供简洁的异步接口

#### 3. 现代化前端应用：

- 基于Vue 3组合式API的响应式架构，组件化程度高，维护性强
- 创新的玻璃拟态UI设计，提供沉浸式的探险游戏体验
- 完整的游戏状态管理，通过computed属性和watch机制实现数据驱动视图

#### 4. 紧密集成的三层架构：

- 前端通过 `Socket.IO` 与后端实时通信，后端通过 `Web3.js` 与区块链网络交互
- 统一的错误处理机制，从前端UI提示到后端日志记录再到区块链交易回执
- 模块化的代码组织，前后端均具备良好的扩展性和可测试性

## 4.3 项目文件

项目源代码结构如下：

- 智能合约代码：/chaincode目录
- 后端代码：/server目录
- 前端代码：/frontend目录
- 项目文档记录：/docs目录

## 4.4 测试结果

- 智能合约测试：所有测试用例通过
- 系统集成测试：各模块协同工作正常
- 用户体验测试：游戏流程顺畅，界面友好

## 5. 改进思路

---

### 5.1 前端界面

- 加入更多图形化显示，增加更具特色的视觉体验效果
- 优化显示性能

### 5.2 游戏机制

- 为游戏引入更丰富的游戏机制，增加游戏可玩性
- 增加更合理的匹配机制

### 5.3 增加MetaMask支持

- 支持用户使用MetaMask账户参与游戏

## 6. 团队组成与分工

---

- 贺禹勋：前端开发
- 邓俊毅：智能合约开发
- 李嘉锐：后端开发

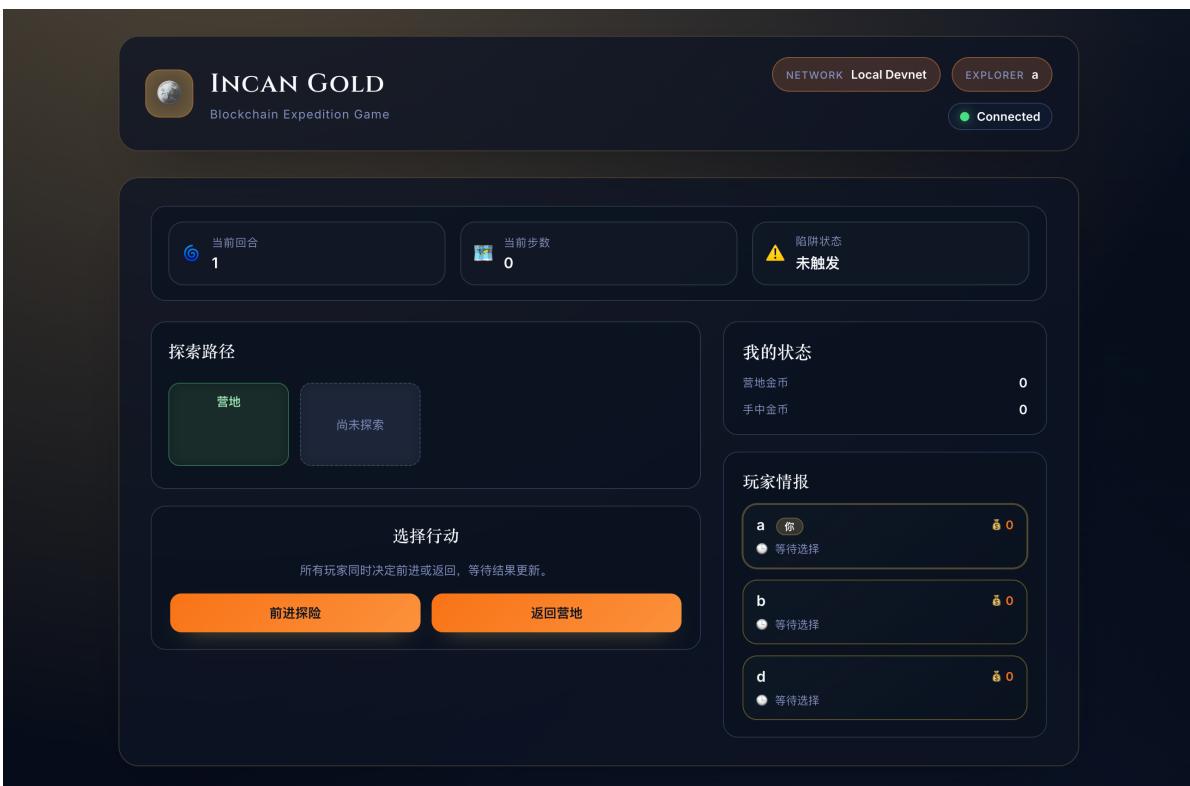
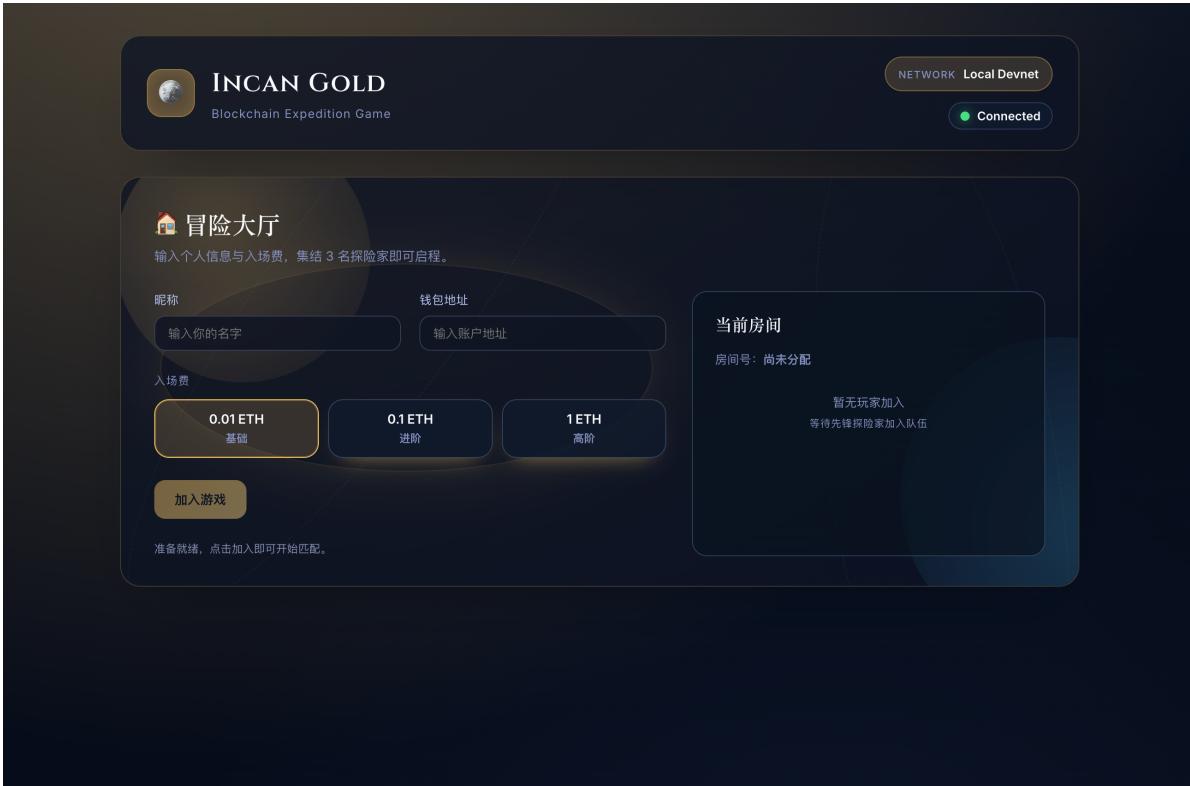
## 7. 附录

---

### 7.1 参考资料

- [以太坊智能合约开发文档](#)
- [Vue.js官方文档](#)
- [Express.js官方文档](#)
- [Socket.io官方文档](#)
- [Web3.js官方文档](#)

### 7.2 项目截图





INCAN GOLD

Blockchain Expedition Game

NETWORK Local Devnet

EXPLORER c

Connected

## 🏆 收益结算

以下为本局链上收益结算，可选择继续冒险或返回大厅。

#1	a 金币 39 投入 0.0100 ETH	0.0175 ETH 净收益 +0.0075 ETH
#2	b 金币 19 投入 0.0100 ETH	0.0085 ETH 净收益 -0.0015 ETH
#3	c 金币 9 投入 0.0100 ETH	0.0040 ETH 净收益 -0.0060 ETH

继续游戏

返回大厅