成都信息工程大學
Chengdu University of Information Technology

本科毕业论文（设计）

数据库设计说明书

学 生 姓 名	翁忠旭
学　　号	2021131028
专　　业	区块链工程
年 级 班 级	2021 级 1 班
指 导 教 师	梁培利（讲师）
所 在 学 院	人工智能学院（区块链产业学院）
提 交 日 期	2025 年 3 月 31 日

2025 年 3 月

成都信息工程大学 人工智能学院（区块链产业学院）

目录

1 引言

1.1 编写目的

在"基于链上随机数和时间锁的彩票系统"项目的前两个阶段，即需求分析和概要设计阶段，已详细阐述系统用户对本系统的需求。本阶段将按照概要设计，对项目的链上数据存储进行详细规划，为开发人员、测试人员和系统维护人员提供参考。

本文档的读者为系统用户，程序开发员，测试人员等。

1.2 背景

1. 本项目的名称为：基于链上随机数和时间锁的彩票系统。

2. 项目背景：传统彩票系统存在中心化问题，导致开奖过程不透明，用户信任度低。 区块链技术的发展使得去中心化彩票系统成为可能。链上随机数和时间锁机制可以确保彩票系统的公平性和透明度。 本项目通过智能合约实现自动化彩票发行、购买、开奖和派奖流程。

1.3 术语

链上随机数：通过区块链技术生成的不可篡改的随机数。
时间锁：基于智能合约的时间约束机制。
智能合约：在区块链上自动执行的程序。
Gas 费用：在以太坊网络上执行交易或合约所需支付的费用。。

1.4 参考资料

[1] Ethereum Foundation. (n.d.). Ethereum Developer Documentation. Retrieved from https://ethereum.org/en/developers/docs/

[2] Hardhat Team. (n.d.). Hardhat Documentation. Retrieved from https://hardhat.org/docs

[3] Drand Team. (n.d.). Drand Documentation. Retrieved from https://drand.love/docs/

[4] Solidity Documentation. (2024). Solidity: Smart Contract-Oriented Programming. Retrieved from https://docs.soliditylang.org/

2 需求分析

2.1 数据存储和处理模式分析

设计目的：

1）需要存储彩票信息、购彩记录和中奖记录等方便查询。

2）降低数据存储成本；

设计数据存储模式：

单一架构（使用区块链网络事件存储和处理部分数据）

3 E-R 模型设计

3.1 实体及属性

系统包括以下主要实体：

1. 彩票(Lottery)

彩票 ID(lotteryId)：主键，唯一标识一个彩票

开始时间(startTime)：彩票销售开始时间

结束时间(endTime)：彩票销售结束时间

开奖时间(unlockTime)：彩票开奖时间

票价(ticketPrice)：单张彩票的价格

奖池总额(totalPrize)：彩票的奖金池总额

中奖号码(winningNumbers)：开奖后的中奖号码

状态(status)：彩票当前状态

2. 用户(User)

用户地址(address)：主键，用户的区块链地址

交易记录(transactions)：用户的交易历史

中奖历史(winningHistory)：用户的中奖记录

3. 彩票购买记录(Ticket)

彩票 ID(ticketId)：主键，唯一标识一张彩票

用户地址(userAddress)：购买者地址

彩票期号(lotteryId)：关联的彩票期号

购买时间(purchaseTime)：购买时间戳

4. 奖项结构(PrizeStructure)

奖项 ID(prizeId)：主键，唯一标识一个奖项

彩票 ID(lotteryId)：关联的彩票 ID

奖项名称(prizeName)：奖项的名称

中奖概率(probability)：中奖概率

奖金比例(prizeRatio)：奖金占奖池的比例

5. 中奖记录(Winner)

中奖 ID(winnerId)：主键，唯一标识一条中奖记录

用户地址(userAddress)：中奖者地址

彩票 ID(lotteryId)：关联的彩票 ID

奖项 ID(prizeId)：中奖的奖项 ID

奖金金额(prizeAmount)：中奖金额

领取状态(claimStatus)：奖金是否已领取

6. 奖金池(PrizePool)

彩票 ID(lotteryId)：主键，关联的彩票 ID

总金额(totalAmount)：奖金池总额

已分配金额(distributedAmount)：已分配的奖金

剩余金额(remainingAmount)：剩余的奖金

7. 交易记录(Transaction)

交易 ID(transactionId)：主键，唯一标识一笔交易

用户地址(userAddress)：交易发起者地址

彩票 ID(lotteryId)：关联的彩票 ID

交易金额(amount)：交易金额

交易时间(timestamp)：交易时间戳

交易类型(transactionType)：交易类型（购买/领奖）

3.2 E-R 图

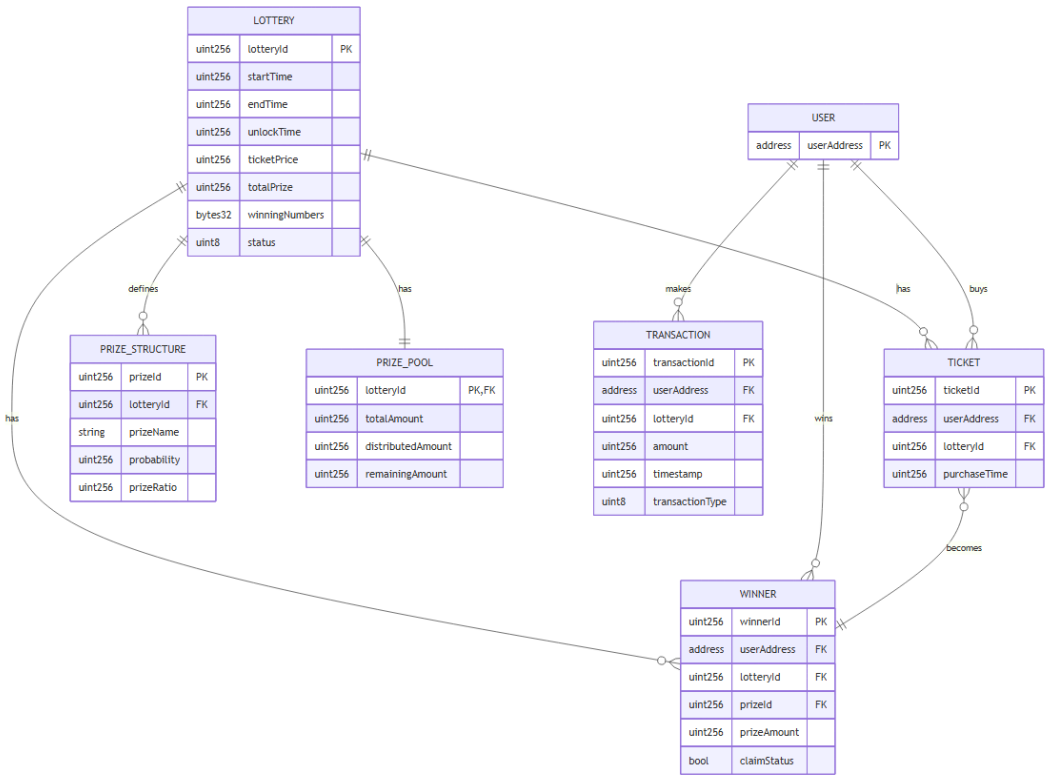

图 3-2 E-R 图

4 数据库实现

4.1 数据库命名约定和环境

4.1.1 命名约定

为确保代码的可读性和一致性，本项目采用以下命名约定：

1. 合约名称：采用大驼峰命名法（PascalCase），如 Lottery、RandomnessProvider

2. 函数名称：采用小驼峰命名法（camelCase），如 createLottery、buyTicket

3. 变量名称：采用小驼峰命名法，如 lotteryId、ticketPrice

4. 常量名称：采用全大写下划线分隔命名法，如 MIN_LOCK_DURATION、MAX_FEE_RATE

5. 事件名称：采用大驼峰命名法，如 LotteryCreated、TicketPurchased

6. 修饰符名称：采用小驼峰命名法，如 onlyOwner、nonReentrant

7. 枚举类型：采用大驼峰命名法，枚举值采用全大写，如 LotteryState.OPEN

8. 结构体名称：采用大驼峰命名法，如 LotteryInfo、PrizeStructure

4.1.2 数据库环境

本项目基于区块链技术，数据存储主要使用链上存储。

4.2 数据库关系图

系统的数据关系图具体内容如图 4-1 所示。

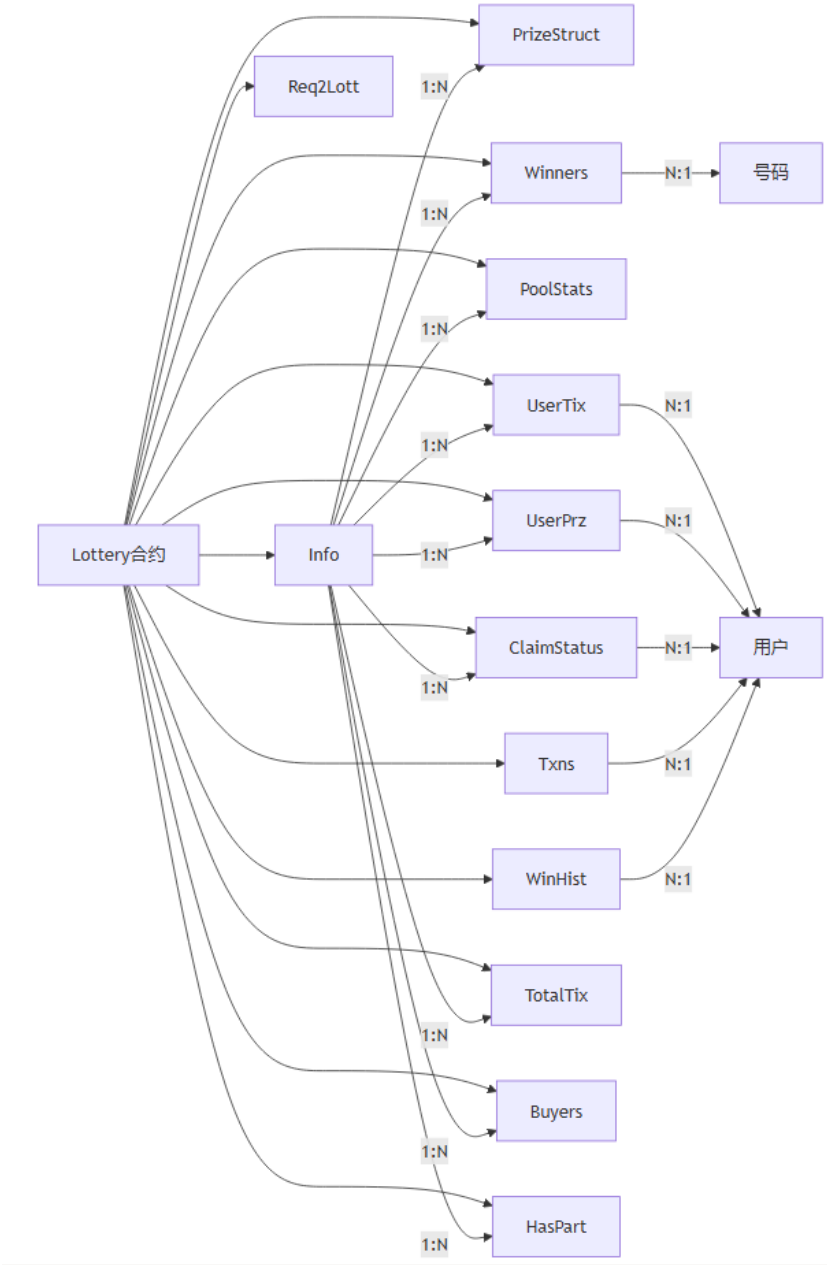

图 4-1 数据关系图

4.3 数据表信息

4.3.1 表列表

表 4-1 表列表

表名	说明	对应合约结构
Lotteries	彩票信息表	mapping(uint256 => LotteryInfo)
PrizeStructures	奖项结构表	mapping(uint256 => PrizeStructure[])
PrizeWinners	中奖记录表	mapping(uint256 => PrizeWinner[])

表名	说明	对应合约结构
UserTickets	用户彩票表	mapping(uint256 => mapping(address => uint256[]))
UserPrizes	用户奖金表	mapping(uint256 => mapping(address => uint256))
PrizeClaimStatus	奖金领取状态表	mapping(uint256 => mapping(address => bool))
UserTransactions	用户交易表	mapping(address => Transaction[])
UserWinningHistory	用户中奖历史表	mapping(address => uint256[])
TotalTickets	彩票总数表	mapping(uint256 => uint256)
LotteryBuyers	彩票购买者表	mapping(uint256 => address[])
HasParticipated	用户参与状态表	mapping(uint256 => mapping(address => bool))
LotteryIdByRequestId	随机数请求映射表	mapping(uint256 => uint256)

4.3.2 具体表结构

4.3.2.1 彩票信息表(Lotteries)

表 4-2 彩票信息表

序号	中文名称	属性名	数据类型	描述
1	彩票 ID	lotteryId	uint256	彩票的唯一标识符
2	开始时间	startTime	uint256	彩票销售开始时间戳
3	结束时间	endTime	uint256	彩票销售结束时间戳
4	开奖时间	unlockTime	uint256	彩票开奖时间戳
5	票价	ticketPrice	uint256	单张彩票的价格（wei）
6	奖池总额	totalPrize	uint256	彩票的奖金池总额（wei）
7	中奖号码	winningNumbers	bytes32	开奖后的中奖号码
8	状态	status	LotteryState	彩票当前状态（枚举）

4.3.2.2 奖项结构表(PrizeStructures)

表 4-3 奖项结构表

序	中文名称	属性名	数据类型	描述

号				
1	奖项 ID	prizeId	uint8	奖项的唯一标识符
2	彩票 ID	lotteryId	uint256	关联的彩票 ID
3	奖项名称	name	string	奖项的名称
4	中奖概率	probability	uint256	中奖概率（百分比）
5	奖金比例	prizeRatio	uint256	奖金占奖池的比例（百分比）

4.3.2.3 中奖记录表(PrizeWinners)

表 4-4 中奖记录表

序号	中文名称	属性名	数据类型	描述
1	中奖 ID	winnerId	uint256	中奖记录的唯一标识符
2	用户地址	winner	address	中奖者的区块链地址
3	彩票 ID	lotteryId	uint256	关联的彩票 ID
4	奖项 ID	prizeRank	uint8	中奖的奖项 ID
5	奖金金额	amount	uint256	中奖金额（wei）
6	中奖时间	winTime	uint256	中奖时间戳

4.3.2.4 用户交易表(UserTransactions)

表 4-5 用户交易表

序号	中文名称	属性名	数据类型	描述
1	用户地址	user	address	用户的区块链地址
2	彩票 ID	lotteryId	uint256	关联的彩票 ID
3	交易金额	amount	uint256	交易金额（wei）
4	交易时间	timestamp	uint256	交易时间戳
5	交易类型	txType	TransactionType	交易类型（枚举）

4.4 存储过程信息

由于区块链智能合约的特性，传统数据库中的存储过程在本系统中以合约函数的形式实现。以下是主要的合约函数：

1. 彩票创建函数

```
function createLottery(
        uint256 _startTime,
        uint256 _endTime,
        uint256 _unlockTime,
        uint256 _ticketPrice,
        PrizeStructure[] memory _prizeStructure
) external onlyOwner returns (uint256)
```

功能：创建新的彩票，设置彩票的基本信息和奖项结构。

2. 彩票购买函数

```
function buyTicket(uint256 _lotteryId) external payable nonReentrant returns (uint256)
```

功能：用户购买彩票，支付相应金额并获得彩票。

3. 彩票开奖函数

```
function drawLottery(uint256 _lotteryId) external onlyOwner returns (bool)
```

功能：触发彩票开奖流程，请求随机数并确定中奖结果。

4. 奖金领取函数

```
function claimPrize(uint256 _lotteryId) external nonReentrant
```

功能：用户领取中奖奖金。

5. 用户中奖历史查询函数

```
function getUserWinningHistory(address _user)
        external
        view
    returns (uint256[] memory lotteryIds, uint256[] memory amounts)
```

功能：获取用户的中奖历史记录。

6. 用户交易历史查询函数

功能：获取用户的交易历史记录。

5 数据库安全设计

5.1 访问控制

角色分配：

系统管理员：具有创建彩票、设置参数等管理权限

普通用户：具有购买彩票、查询信息、领取奖金等权限

权限控制：

使用 OpenZeppelin 的 Ownable 合约实现管理员权限控制

使用修饰符(modifier)限制特定函数的访问权限

关键操作(如创建彩票、开奖)只能由管理员执行

5.2 数据安全

重入攻击防护：

使用 OpenZeppelin 的 ReentrancyGuard 合约防止重入攻击

在涉及资金转移的函数中使用 nonReentrant 修饰符

整数溢出保护：

使用 Solidity 0.8.x 版本内置的溢出检查

关键计算使用 SafeMath 库进行额外保护

随机数安全：

使用外部随机数服务(drand)获取可验证的随机数

避免使用区块哈希等可被矿工操纵的随机源