中山大学数据科学与计算机学院本科生实验报告

(2019 年秋季学期)

课程名称：区块链原理与技术　　任课教师：郑子彬

年级	2017 级	专业（方向）	软件工程
学号	17343077	姓名	刘露
电话	15626443016	Email	1501699757@qq.com
开始日期	2019/10/22	完成日期	2019/12/13

一、 项目背景

➢ 区块链+供应链金融：

将供应链上的每一笔交易和应收账款单据上链，同时引入第三方可信机构来确认这些信息的交易，例如银行，物流公司等，确保交易和单据的真实性。同时，支持应收账款的转让，融资，清算等，让核心企业的信用可以传递到供应链的下游企业，减小中小企业的融资难度。

➢ 实现功能：

功能一：实现采购商品—签发应收账款 交易上链。例如车企从轮胎公司购买一批轮胎并签订应收账款单据。

功能二：实现应收账款的转让上链，轮胎公司从轮毂公司购买一笔轮毂，便将于车企的应收账款单据部分转让给轮毂公司。轮毂公司可以利用这个新的单据去融资或者要求车企到期时归还钱款。

功能三： 利用应收账款向银行融资上链，供应链上所有可以利用应收账款单据向银行申请融资。

功能四：应收账款支付结算上链，应收账款单据到期时核心企业向下游企业支付相应的欠款。

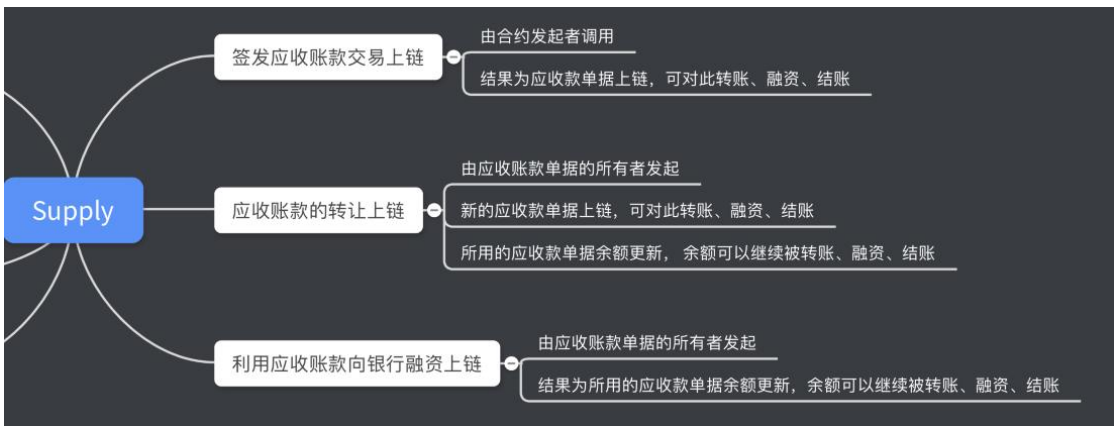
二、 方案设计

➢ 数据流图示

➢ 储存设计

前端为表格存储，后端存储相关信息到日志中，链端则负责交易上链。

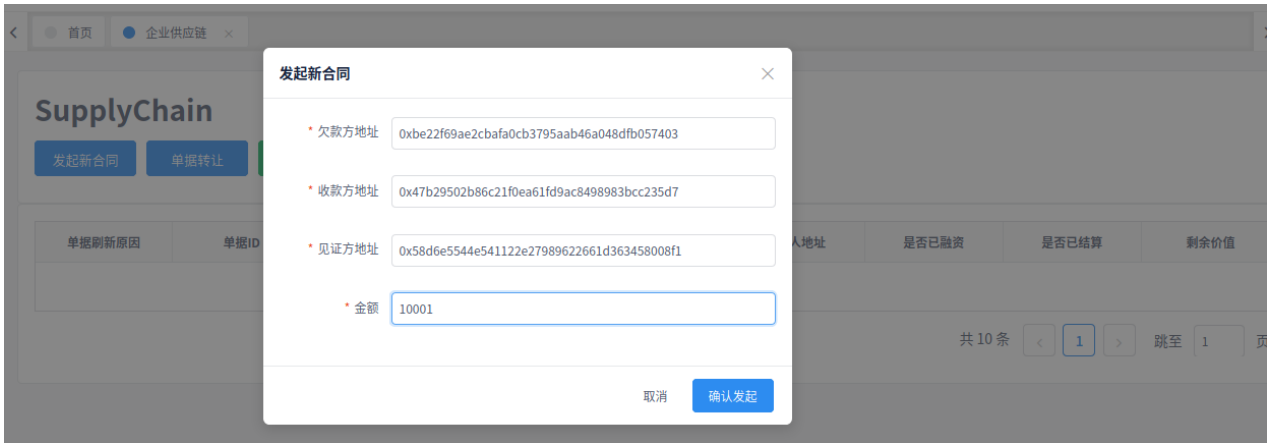
➢ 核心功能思路



三、 功能测试

➢ 功能一、发起合同

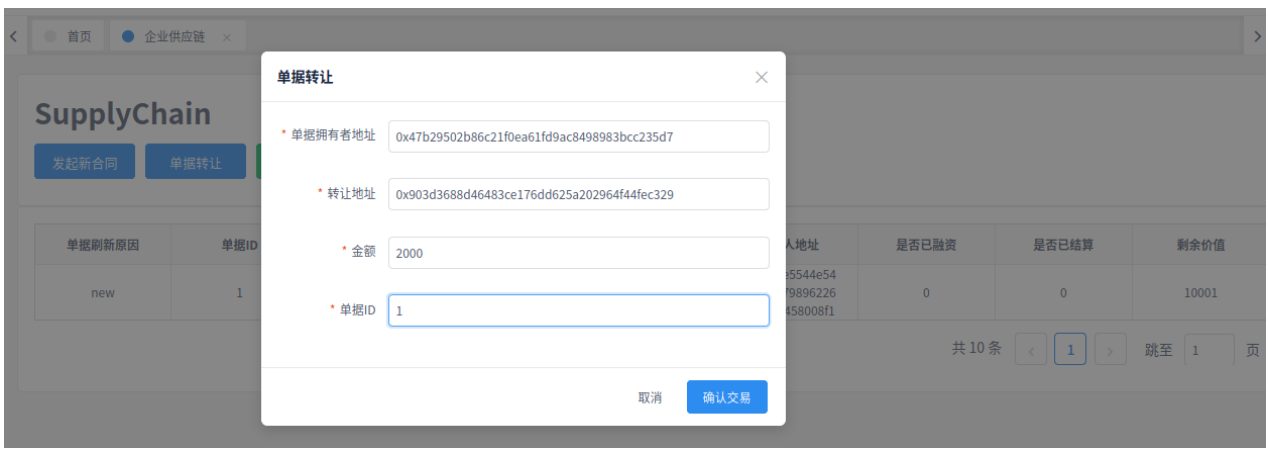
填表



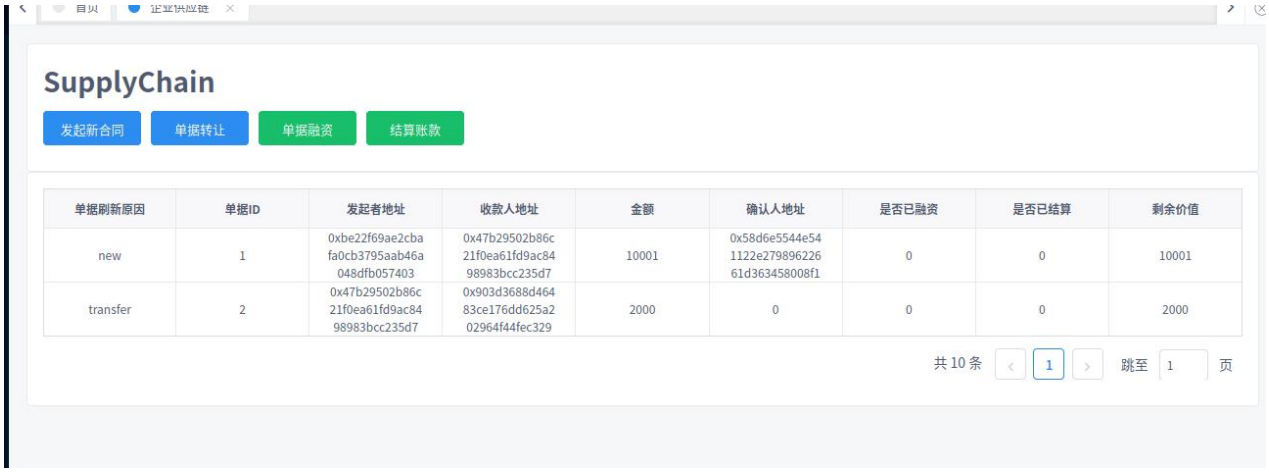
输出

➢ 功能二、单据转让

填表



输出



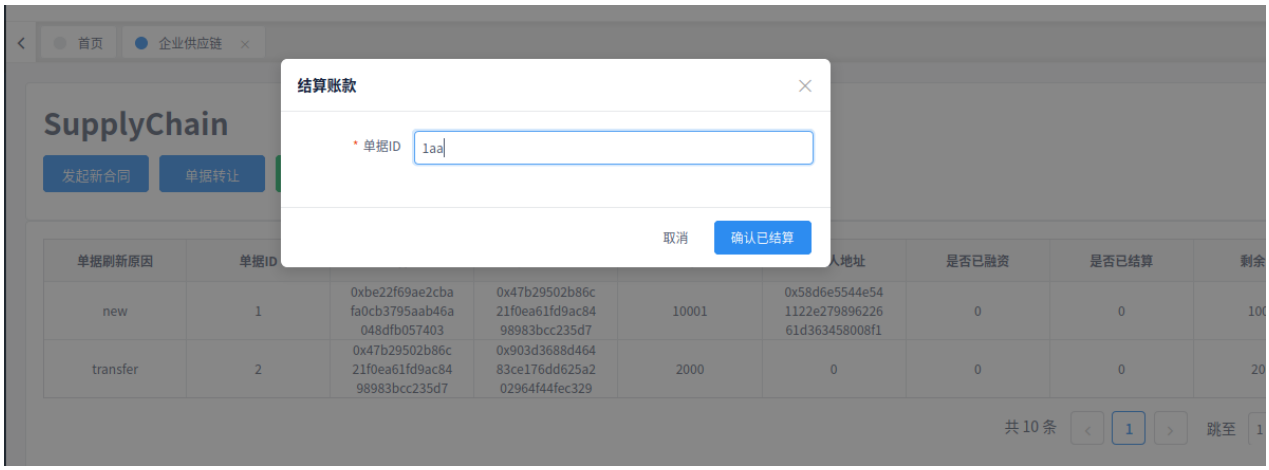
➢ 功能三、单据融资

填表

输出

➢ 功能四、结算账款

填表

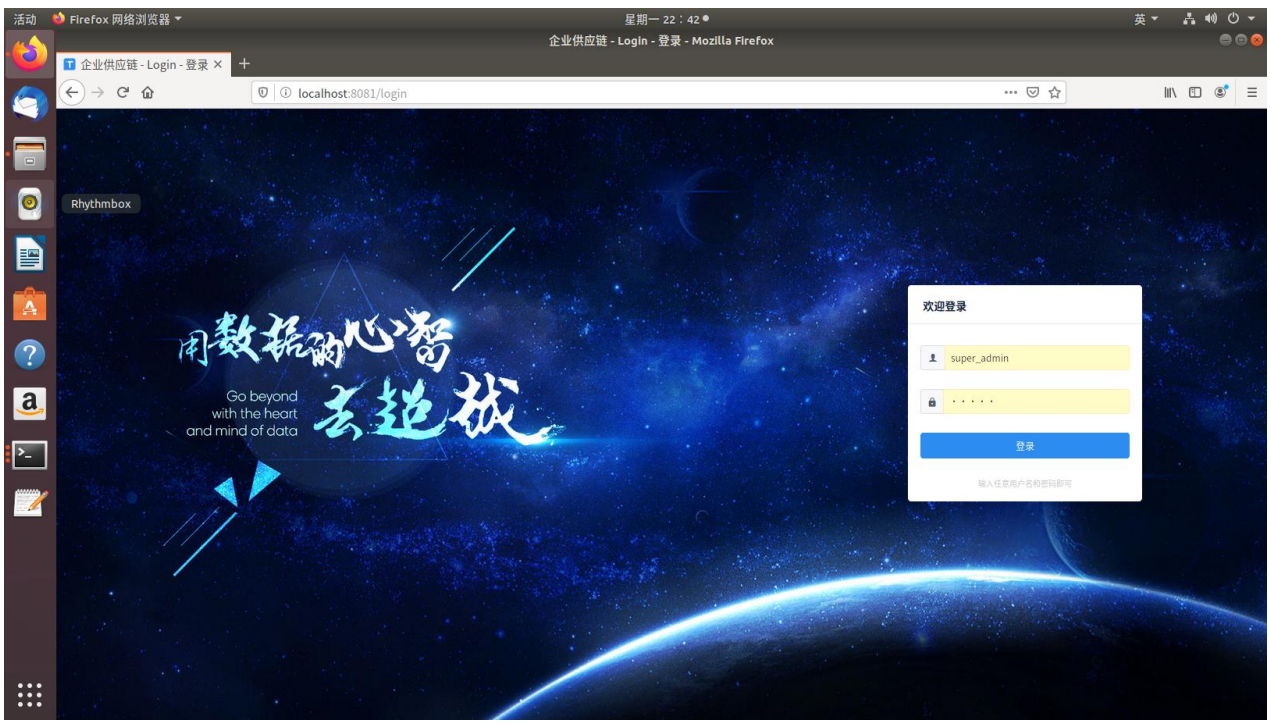


输出

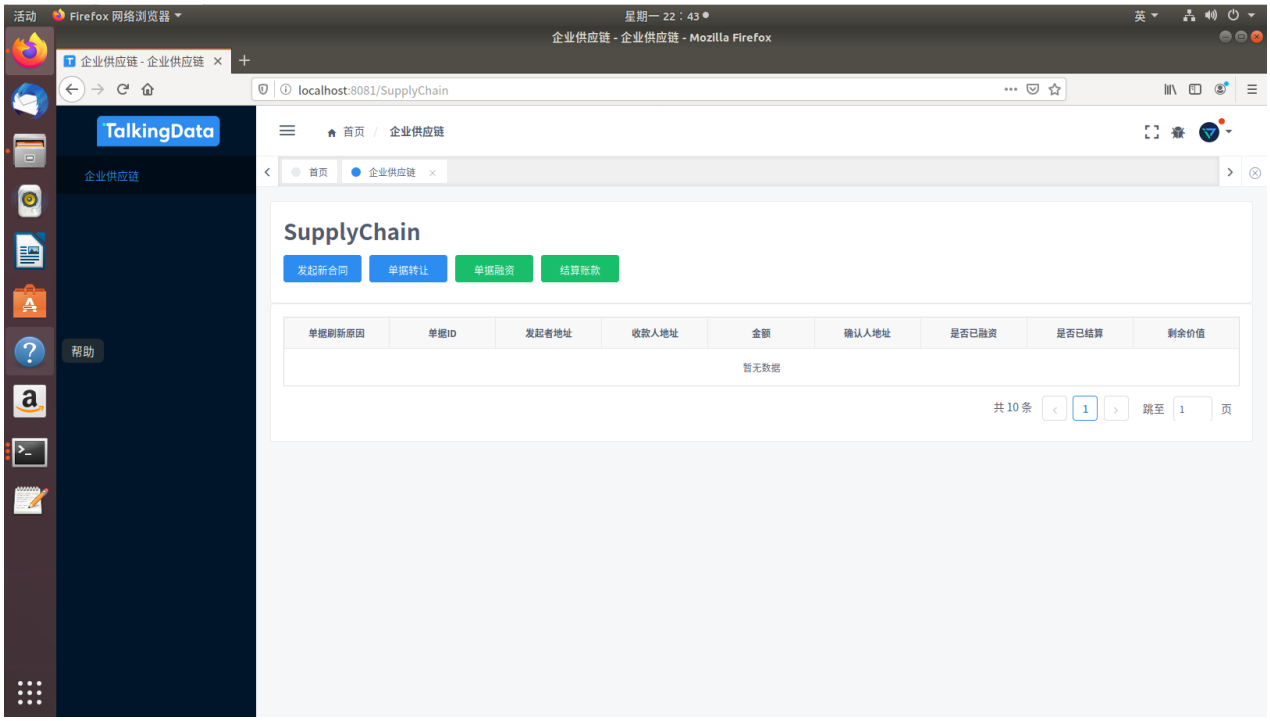
```
  func: 'Financing',
  leftValue: 0 }

aatemp { receiptID: '1aa', func: 'Settle' }
```

四、 界面展示

➢ 登录界面



➢ 表格界面



➢ 框架中控制页面主要代码（src/view/eouter.）

```
import Main from '@/components/main'
import parentView from '@/components/parent-view'

/**
 * iview-admin 中 meta 除了原生参数外可配置的参数:
 * meta: {
 *  title: { String|Number|Function }
 *         使用'{{ 多语言字段 }}'形式结合多语言使用，例子看多语言的路由配置;
 *         可以传入一个回调函数，参数是当前路由对象，例子看动态路由和带参路由
 *  hideInBread: (false) 设为 true 后此级路由将不会出现在面包屑中，示例看 QQ 群路由配置
 *  hideInMenu: (false) 设为 true 后在左侧菜单不会显示该页面选项
 *  notCache: (false) 设为 true 后页面在切换标签后不会缓存，如果需要缓存，无需设置这个字段，而且需要设置页面组件 name 属性和路由配置的 name 一致
 *  access: (null) 可访问该页面的权限数组，当前路由设置的权限会影响子路由
 *  icon: (-) 该页面在左侧菜单、面包屑和标签导航处显示的图标，如果是自定义图标，需要在图标名称前加下划线'_'
 *  beforeCloseName: (-) 设置该字段，则在关闭当前 tab 页时会去'@/router/before-close.js'里寻找该字段名对应的方法，作为关闭前的钩子函数
 * }
 */

export default [
  {
    path: '/login',
    name: 'login',
    meta: {
```

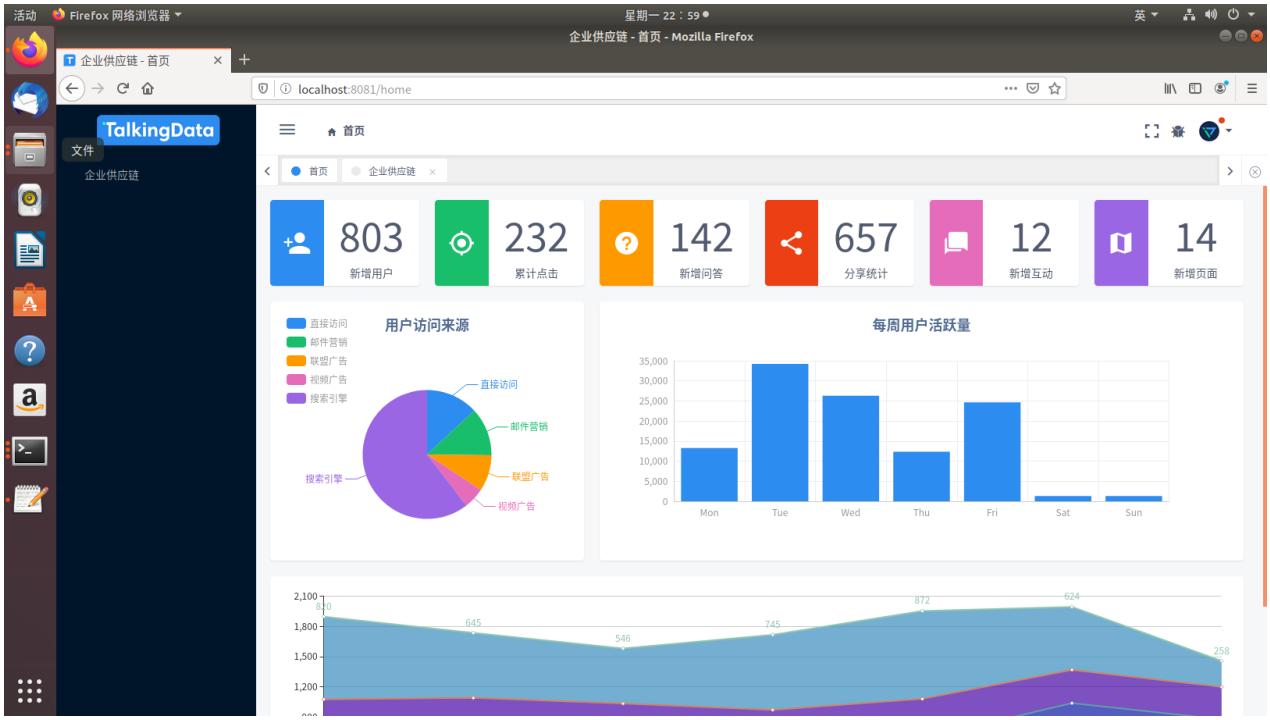
```
    title: 'Login - 登录',
    hideInMenu: true
  },
  component: () => import('@/view/login/login.vue')
},
{
  path: '/',
  name: '_home',
  redirect: '/home',
  component: Main,
  meta: {
    hideInMenu: true,
    notCache: true
  },
  children: [
    {
      path: '/home',
      name: 'home',
      meta: {
        hideInMenu: true,
        title: '首页',
        notCache: true,
        icon: 'md-home'
      },
      component: () => import('@/view/single-page/home')
    }
  ]
},

/*
//发起合同
{
  path: '/Receipt',
  name: 'Receipt',
  component: Main,
  meta: {
    hideInBread: true
  },
  children: [
    {
      path: '/Receipt',
      name: '发起合同',
      meta: {
        // icon: '_qq',
        title: '发起合同'
      },
      component: () => import('@/view/Receipt.vue')
    }
  ]
},

//单据转让
{
  path: '/Transfer',
  name: 'Transfer',
```

```
  component: Main,
  meta: {
    hideInBread: true
  },
  children: [
    {
      path: '/Transfer',
      name: '单据转让',
      meta: {
        // icon: '_qq',
        title: '单据转让'
      },
      component: () => import('@/view/Transfer.vue')
    }
  ]
},

//单据融资
{
  path: '/Financing',
  name: 'Financing',
  component: Main,
  meta: {
    hideInBread: true
  },
  children: [
    {
      path: '/Financing',
      name: '单据融资',
      meta: {
        // icon: '_qq',
        title: '单据融资'
      },
      component: () => import('@/view/Financing.vue')
    }
  ]
},

//结算账款
{
  path: '/Settle',
  name: 'Settle',
  component: Main,
  meta: {
    hideInBread: true
  },
  children: [
    {
      path: '/Settle',
      name: '结算账款',
      meta: {
        // icon: '_qq',
        title: '结算账款'
      },
```

```
        component: () => import('@/view/Settle.vue')
      }
    ]
  },
  */

 {
  path: '/SupplyChain',
  name: 'SupplyChain',
  component: Main,
  meta: {
    hideInBread: true
  },
  children: [
    {
      path: '/SupplyChain',
      name: '企业供应链',
      meta: {
        // icon: '_qq',
        title: '企业供应链'
      },
      component: () => import('@/view/SupplyChain.vue')
    }
  ]
},

  {
    path: '/message',
    name: 'message',
    component: Main,
    meta: {
      hideInBread: true,
      hideInMenu: true
    },
    children: [
      {
        path: 'message_page',
        name: 'message_page',
        meta: {
          icon: 'md-notifications',
          title: '消息中心'
        },
        component: () => import('@/view/single-page/message/index.vue')
      }
    ]
  }
]
```

五、 加分项

使用了 iview admin这个框架，有一个良好的用户界面:



六、 心得体会

本次大项目让我收获很多，从一开始的实验一私有链的搭建到实验二项目设计，再到最后实验三最终制品的成功设置，循序渐进，让我从从思考到设计再到实现，一步一步完成最后的项目，很有成就感，每一步都学习到了很多的相关知识。

实验一的热身比较简单，让我对开源区块链系统 FISCO-BCOS和其私有链的搭建有了基本的了解，对于后面的实验是不可缺少的知识。

实验二主要在于项目的设计和合约的编写，在一开始觉得有点摸不着头脑。但通过上网查找资料、与同学的讨论和自己的摸索，慢慢明白了该实验要实现的内容（实质就是一些函数的实现），便也觉得不算太难，唯一的问题就是不同版本编辑器的代码相差还有点大，我在学习时（主要是学习网上已有的代码）有点混乱。

在完成实验三的过程中，我则遇到了许多的难题。

第一个就是不知道从何开始。在链端和后端的链接中，我一开始只懂调用，不懂修改，看了官方的文档也还是很懵，就算是在群里问了同学和研究人员也还是不明不白，

原因就在于我并没有懂clil的实质，只知道cli的输入输出而没有去了解Nodejs JDK的真正的实现理论，最后浪费了很多时间去包装cli的输出之后才明白这个道理。然后我才去研究Nodejs JDK的源码，之后豁然开朗，总算知道了我应该干嘛。

其实在前端的过程中我也碰到了和第一个类似的问题，我选用了 iview admin 这个Nodejs的框架，却很拒绝看官方文档而选择去百度，结果可想而知，由于并不知道框架的实际思想，一开始只会照猫画虎，后面总算明白了还是得看源码啊！

还有一个特别麻烦的事情就是项目中的实现有时候需要改掉原有已经编译好没问题的代码，所以在此过程中我常常会忘记保存之前的版本而循环往复地对一个代码的一个问题来来回回地进行查找问题，所以及时地对项目进行备份并备注真的很重要！

另外，还有一些由于不够仔细和心态不好造成的问题。其中印象深刻的一次就是我忘记关掉端口而导致后面的代码无法运行，当时桌面又开了特别多的界面（原因就是为了保存版本结果又太多太多了很混乱），自己心态又很慌，以为是代码有bug，结果一直找了好久好久，最后才发现不是代码的问题，这些小小的错误，却导致了大把时间的浪费，可见细心是多么重要。

总的来说，通过这次实验，我对区块链基础知识、nodejs的前后端分离项目的实现、框架的学习与运用都有了很多的收获！

而这次实验最重要的是，因为在实验中碰到了很多的困难，我有无数次想要放弃，但最终还是选择了坚持，并在坚持的过程中找到其中的一些乐趣，我不仅仅学到了东西，更让我锻炼了意志（不仅仅是debug，虚拟机的速度也真的很锻炼我），也对项目和区块链有了更多的学习乐趣。希望在以后的日子里，我也能继续加油。

作业要求：

1. 命名要求: 学号_姓名，例如 16340000_林 XX。
2. 实验报告提交格式为 pdf。
3. 实验内容不允许抄袭，我们要进行代码相似度对比。如发现抄袭，按 0 分处理。