# Medical Record

## 选题背景

当前社会上每个人基本上都有多个病历本，分别由不同的医院掌握，

当我们去看病时，医院可能没有或没有完全掌握我们的病史情况和医疗记录，这

样就会给我们的就医造成很大的困扰，有时甚至会导致误诊，不能对症下药。

## 选题依据

如果能够利用区块链技术来保存我们的病历，就能保证我们每个人都

只有一个病历本，并且由我们自己掌握，需要时再由用户授权给医院查询特定的

病史和医疗记录，以及添加新的就医记录，同时限制医院不能更改已有的医疗记

录。这样既能保护用户的隐私，又能保证病历的不可篡改性，还能使用户未来看

病或对自己健康做规划时有数据可用。

## 实验步骤

**环境搭建**

1. 安装Node（官网下载）

2. 安装Truffle（npm install –g truffle）

3. 安装web3（npm install web3）

4. 安装Ganache（官网下载）

5. 安装lite-server（npm install –g lite-server）

6. 安装MetaMask（浏览器扩展程序）

**创建项目**

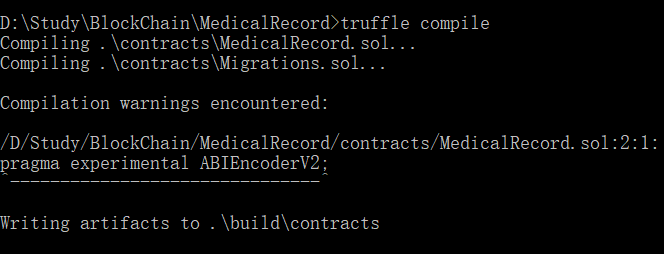
truffle init 创建全新的项目

**编写智能合约**

在contracts目录下，添加合约文件MedicalRecord.sol

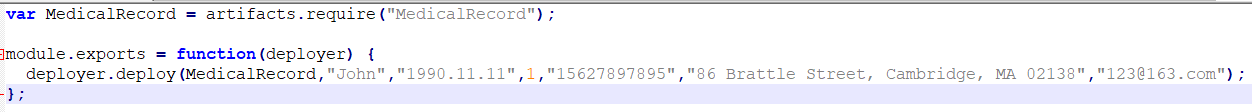
**编译智能合约**

truffle compile

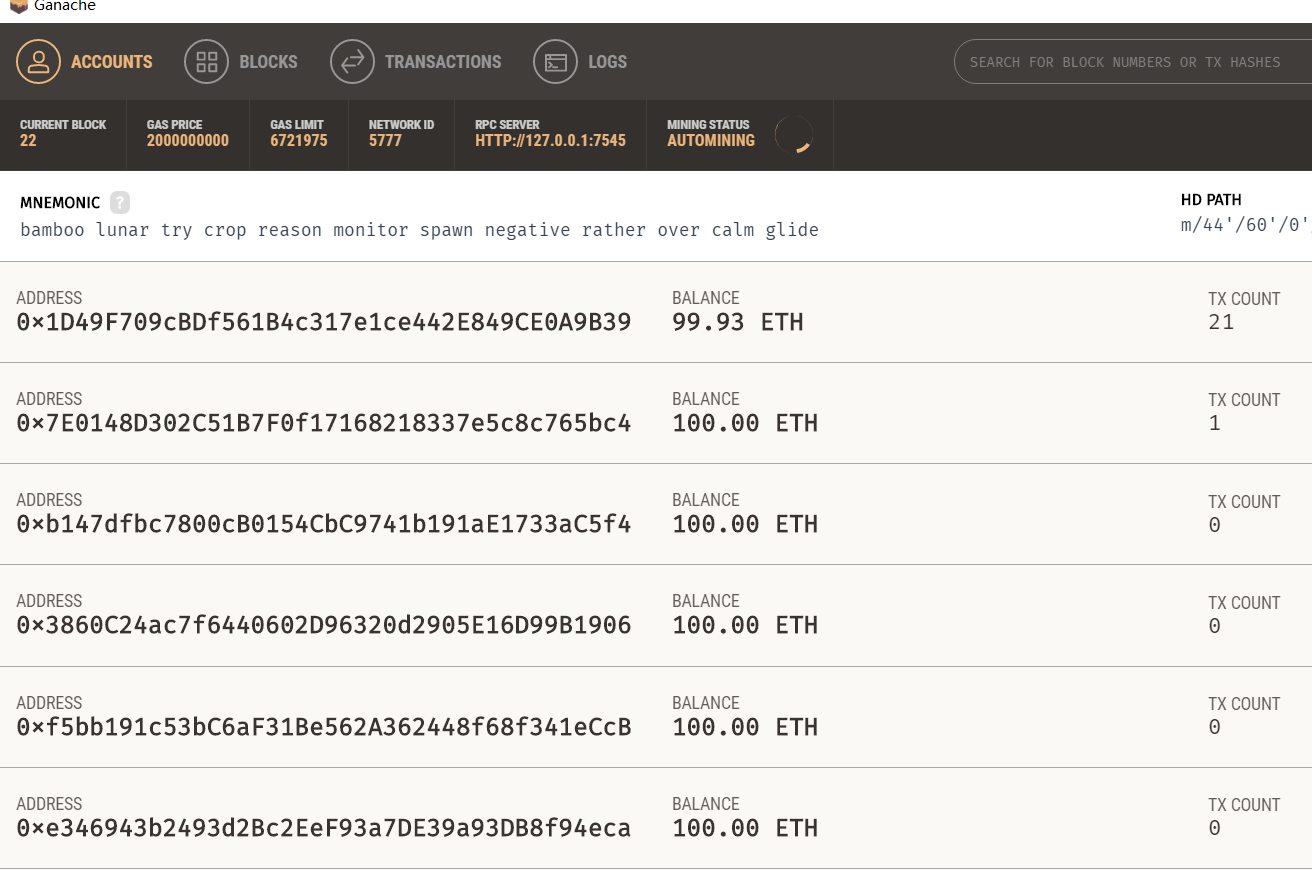


**部署智能合约**

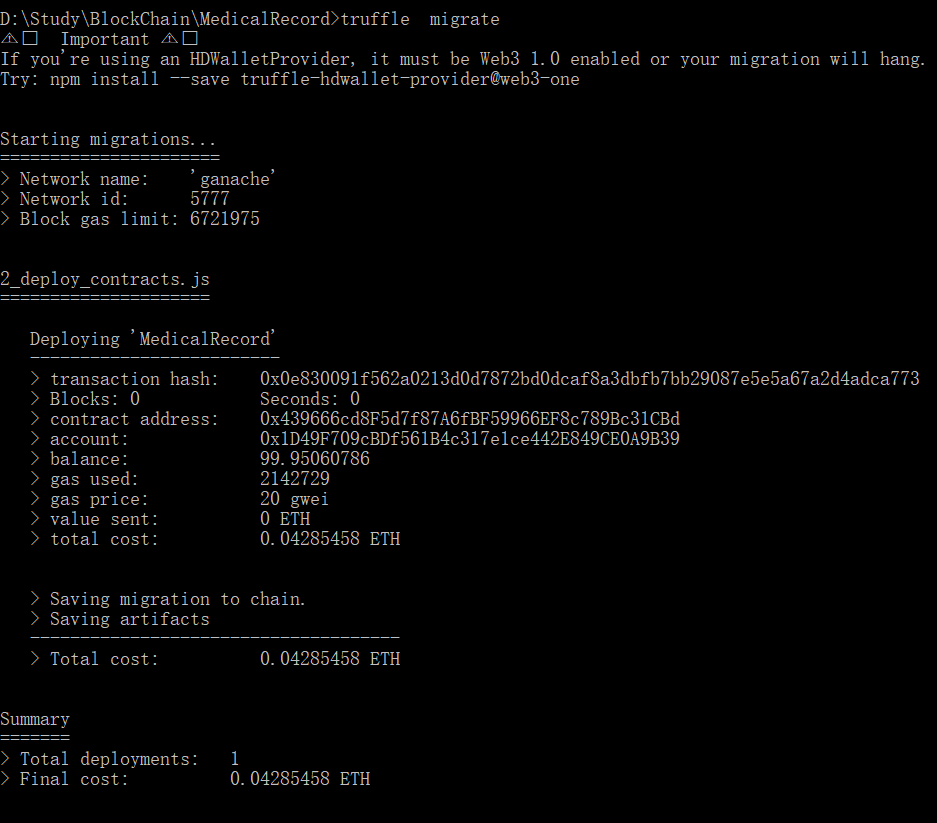
1. 在migrations文件夹下创建部署脚本2\_deploy\_contracts.js

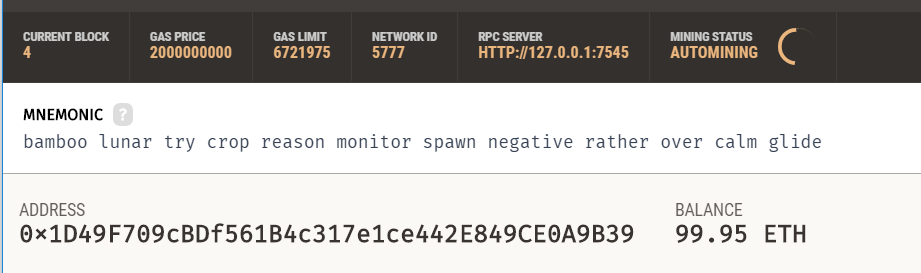


2. 启动Ganache来开启一个私链进行开发测试



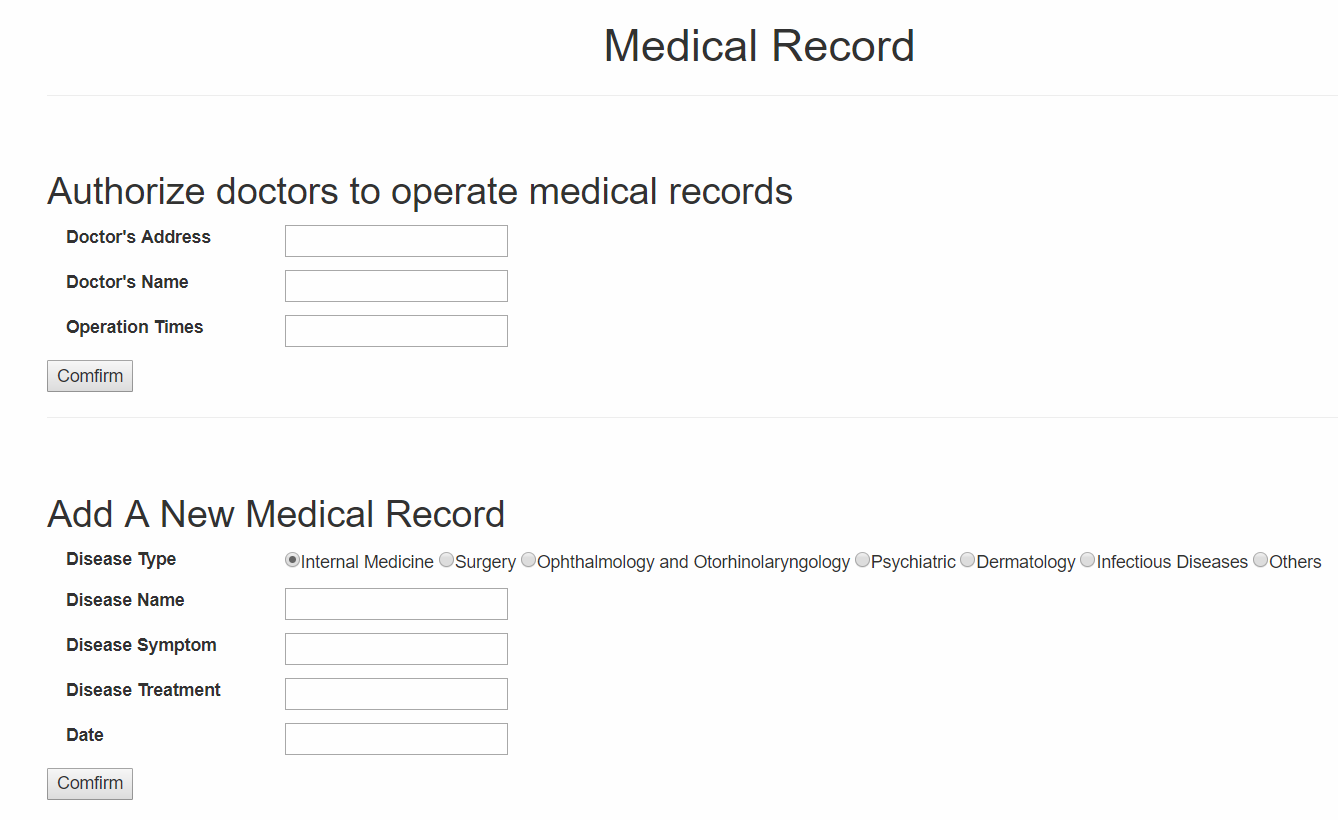
truffle migrate执行部署命令

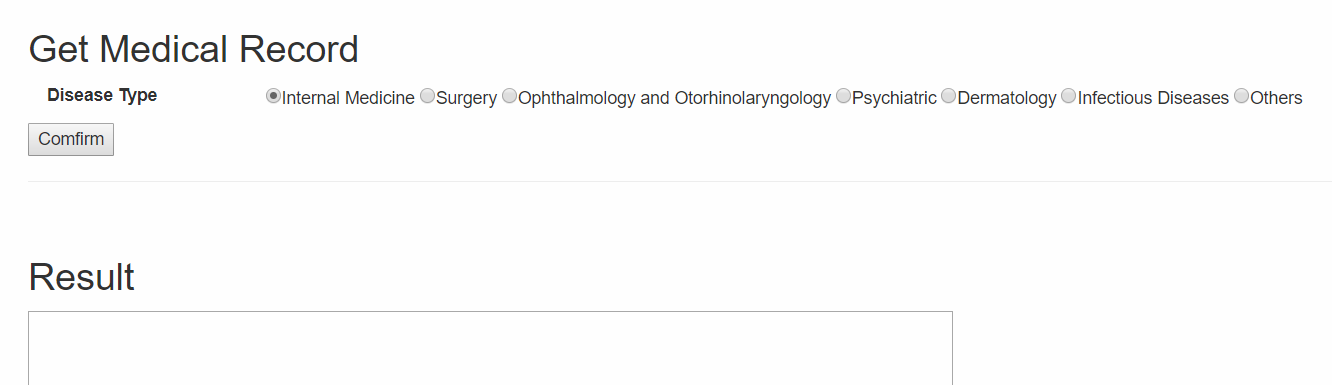




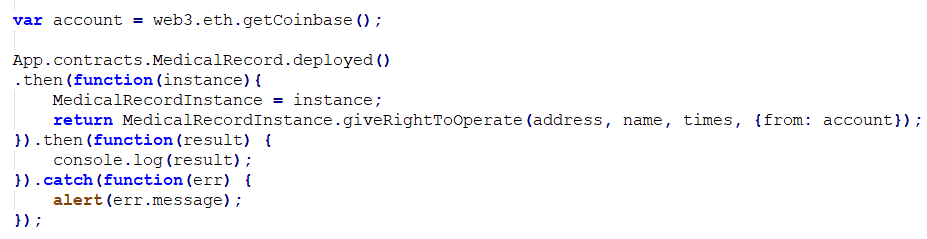
产生了四个区块说明部署成功

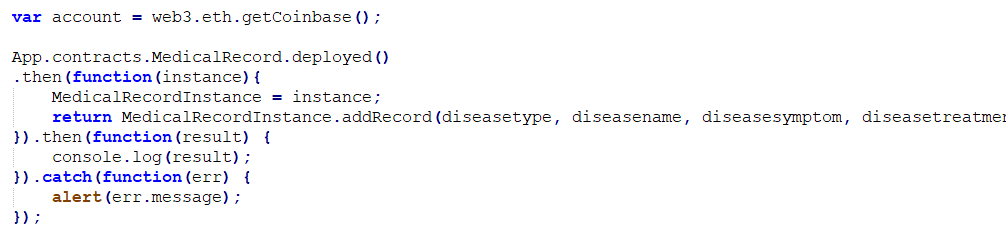
**编辑HTML文件创建布局**

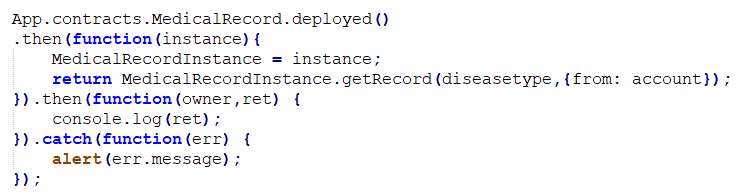




**创建用户接口和智能合约交互**







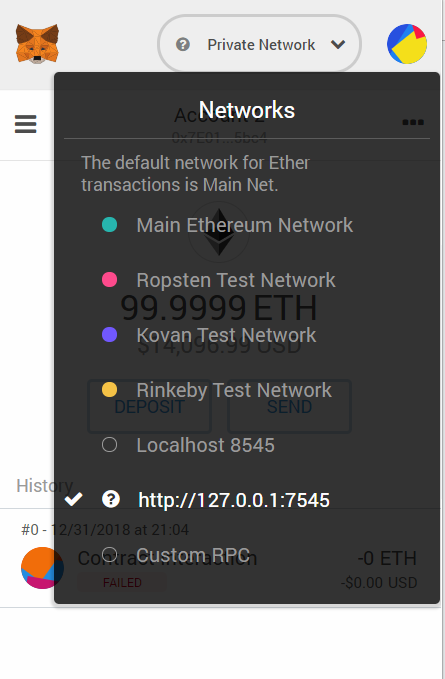
**运行测试**

复制Ganache助记词



输入自定义密码进入MetaMask

选择Custom RPC添加新网络[**http://127.0.0.1:7545**](http://127.0.0.1:7545/)



配置lite-server

bs-config.json指示了lite-server的工作目录。

{  
 "server": {  
 "baseDir": ["./src", "./build/contracts"]  
 }  
}

package.json文件的scripts中添加了dev命令

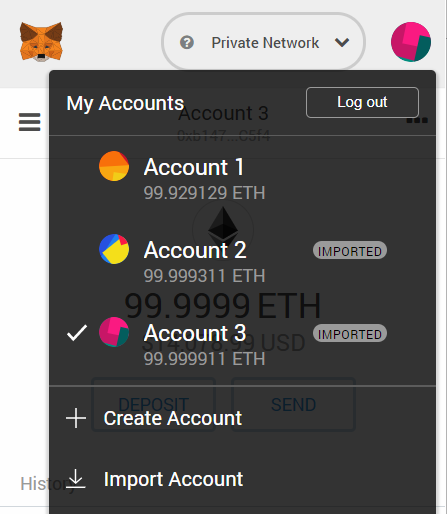
"scripts": {  
 "dev": "lite-server",  
 "test": "echo \"Error: no test specified\" && exit 1"  
},

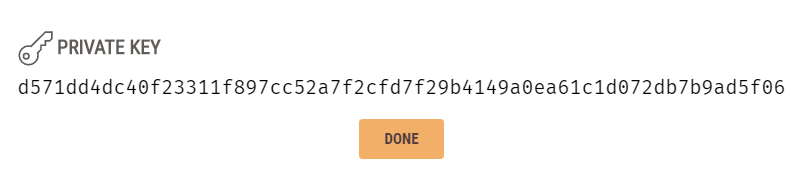
使用指令npm run dev启动lite-server运行并测试DAPP

## 使用说明

1. 通过MetaMask可以实现多个账户对合约交互，合约拥有者为Account 1

添加账户可以点击右上角，Import Account，复制Ganache中其中一个用户的密钥并粘贴



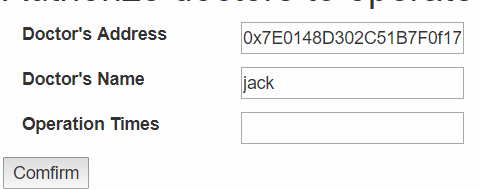


2. 使用Account 1可以给其他账户授权操作他的病历

3. Account 1和被授权的账户可以查询和添加Account 1的病历记录

## 测试

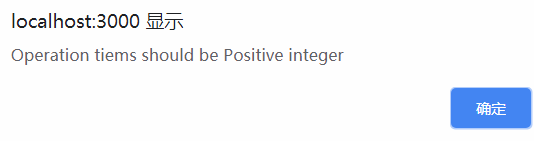
输入存在空



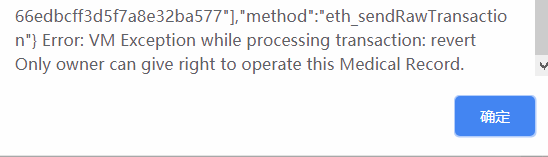


操作次数不为正整数

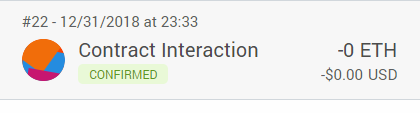
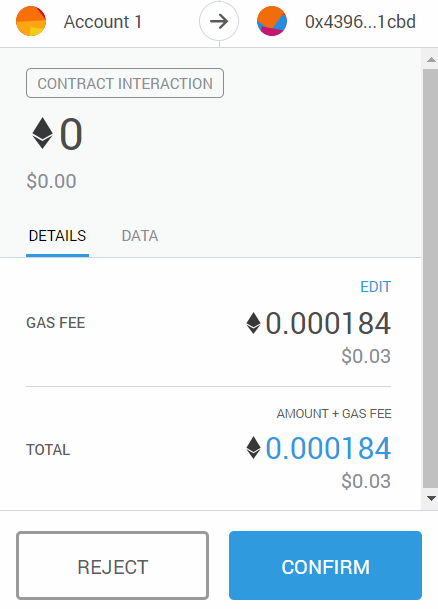




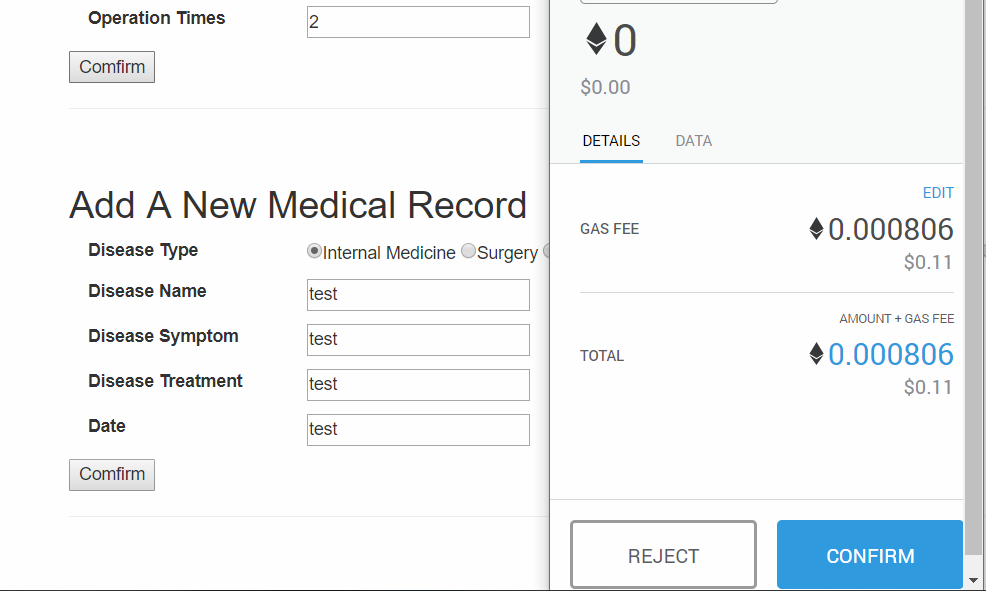
授权者不是合约的持有者

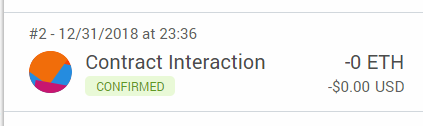


正常授权

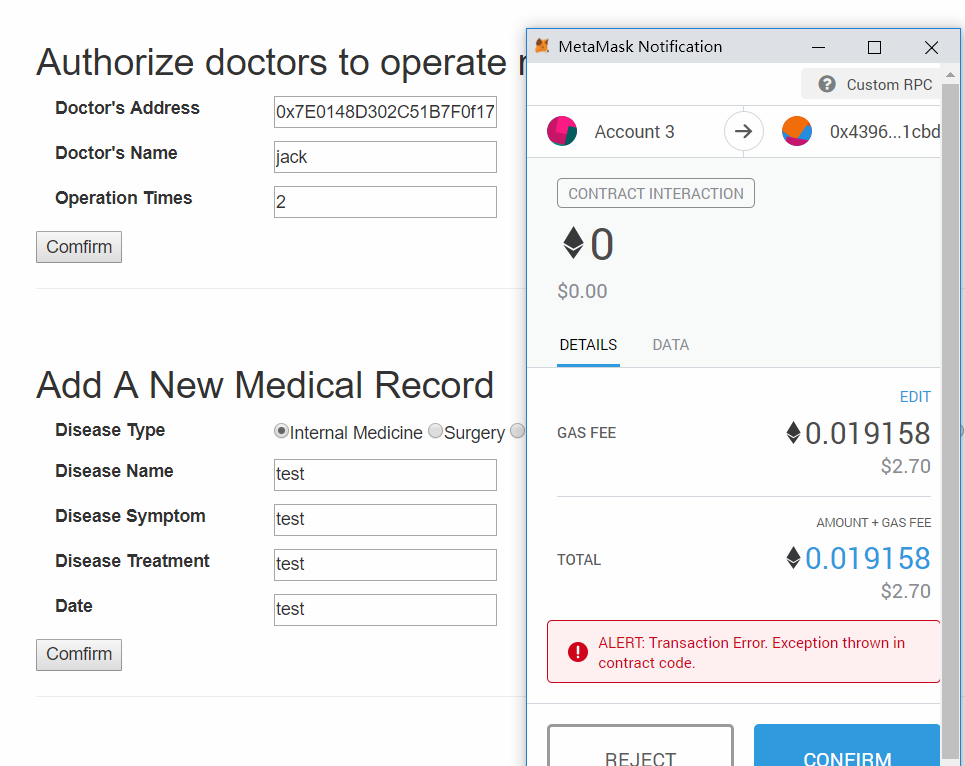


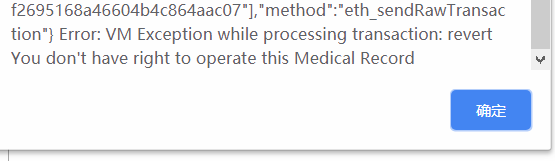
授权用户添加记录





未授权用户添加记录





获取病历记录还存在一点问题，改进和测试见Github

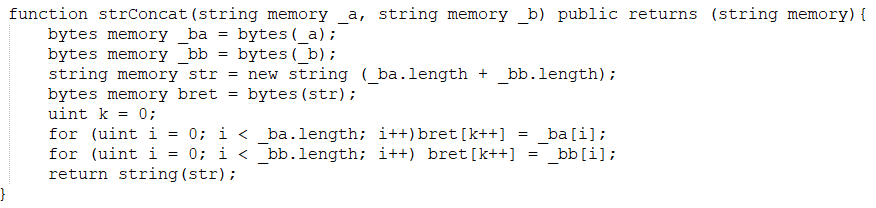
Github地址：<https://github.com/Zhanggen-sysu/MedicalRecord>

参考资料：[https://learnblockchain.cn/2018/01/12/first-dapp/#more](https://learnblockchain.cn/2018/01/12/first-dapp/%23more)

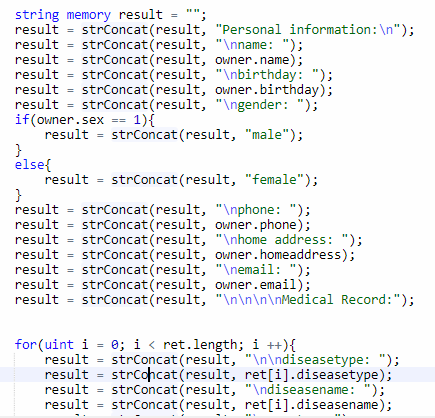
## 报告补充

由于之前合约查询返回的是一个结构体和一个结构体数组，处理极为复杂，所以修改了合约，将返回变成一个字符串，但最后的效果是一样的。

Solidity语言不支持加法字符串拼接，所以手动实现：



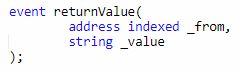
返回字符串的代码如下



使用return 返回，返回的是交易的Hash，所以修改为利用事件返回：

参考资料：<https://www.tuicool.com/articles/J7JbUvq>

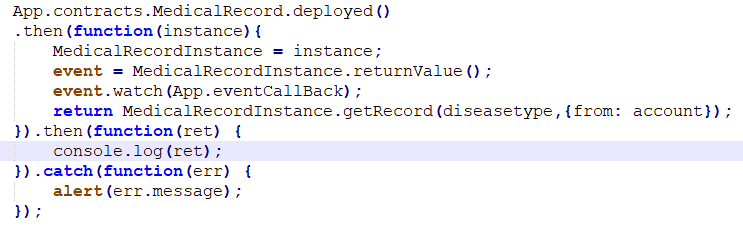
定义事件

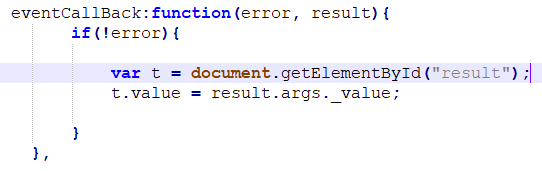


触发事件



在js中：

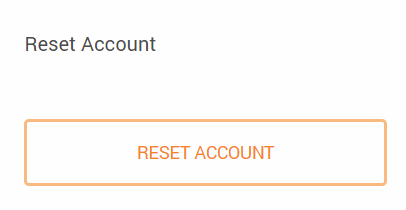




利用回调函数更新UI  
**其他补充**

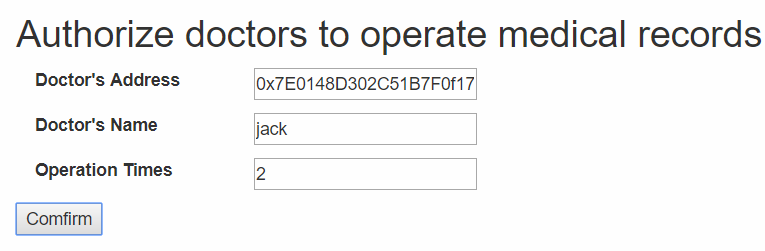
重新编译合约后再次部署需要truffle network –clean清理原来的部署

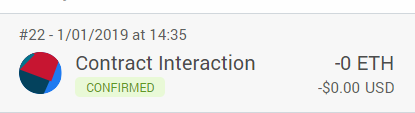
使用MetaMask出现nonce错误需要先重置账户

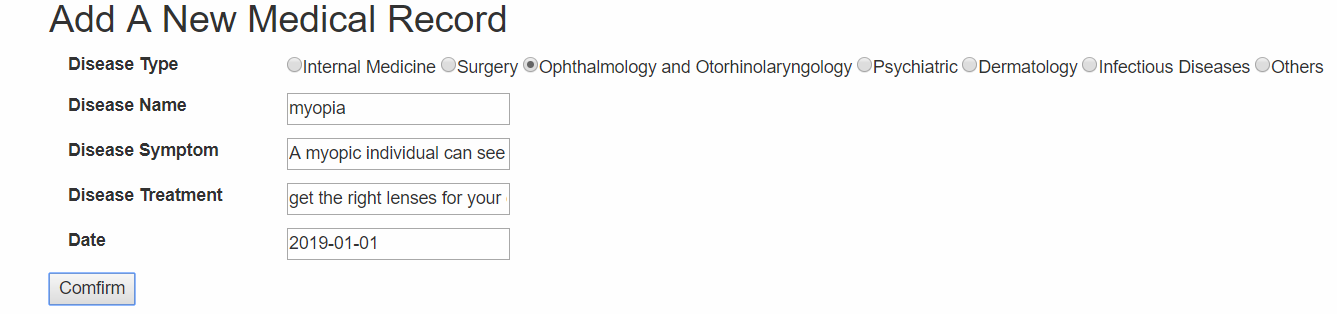


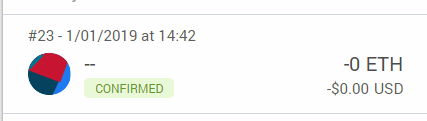
Invalid address可以通过刷新页面解决

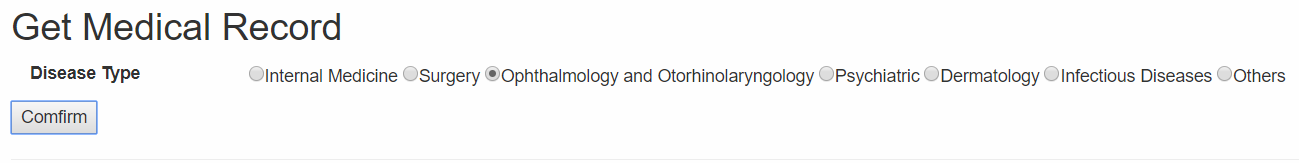
## 测试补充



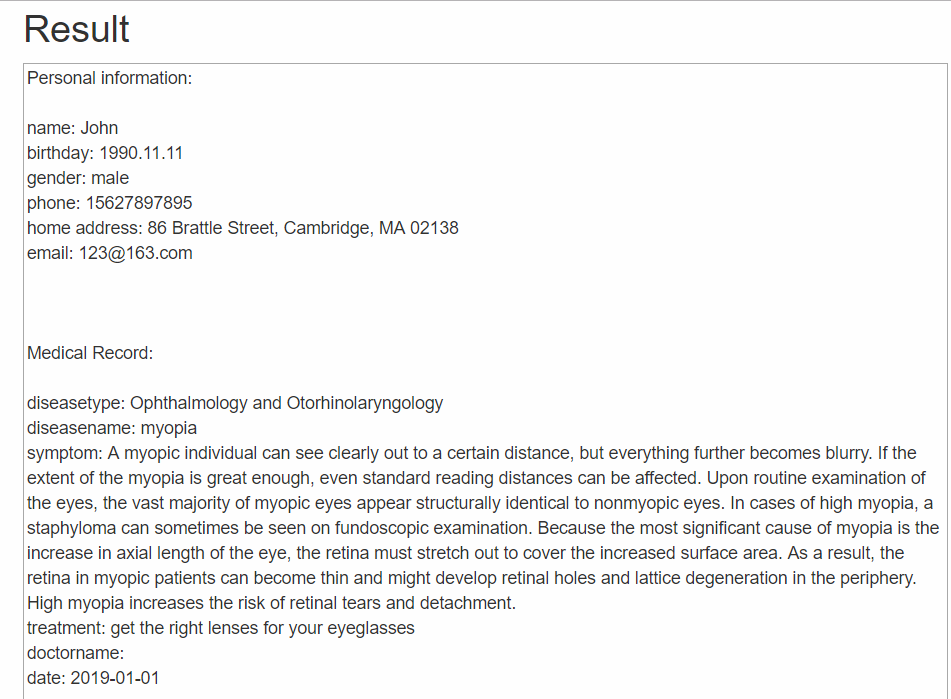








自己添加的所以doctorname为空



使用Account 2添加并查询

