程序仅考虑医生和患者之间的数据交互, 医院在这里只是充当一个被动的数据库, 它配合区块链, 维护患者数据的索引信息, 并通过RESTful API向用户(医生和患者)提供数据访问接口.
医院(数据库)暂且不考虑数据的产生问题

下面都是站在区块链的角度看待整个系统

医生向区块链发起请求, 请求获得病人的数据:

主要流程:

```
医生发起请求  ->  患者同意请求  ->  医院数据库为该访问请求生成索引记录  ->  医生可以访问医院数据库中指定的患者数据
```

request(doctor, patient)对应的情况是医生在发起请求时暂时还没拿到患者的许可, 这种情况下, 区块链只是记录下医生发起的请求, 只有在后续获得患者同意的情况下, 才会通知医院为该请求生成数据索引. 同时, 区块链会触发一个事件, 用来通知对应的患者有新的请求. 患者此时会收到通知, 来决定是否同意医生的请求. 当然, 这里并不要求患者必须上线, 即便患者此时并不在线, 医生所发起的请求也不会消失, 请求会被区块链记录下来, 只不过医生在没拿到患者许可的情况下无法访问数据而已.

区块链储存的信息包括:

```
"request" doctor patient txid  ->  txTimestamp
"doctorLastRequest" doctor  ->  N
txid  ->  request $$ doctor $$ patient $$ txTimestamp
"patient" patient txid  ->  doctor $$ txTimestamp
```

基于第一条记录, 可以
- 查看某个医生所发起的所有请求
- 查看某个医生针对某个患者所发起的所有请求
- 查看某个医生针对某个患者所发起的某一个特定的请求

第二条记录是用来确保医生的请求是合法的, 详情见后面的faq1
基于第三条记录, 可以
- 查看某条请求的详细信息

基于第四条记录, 可以
- 查看关于某个患者的所有或某一条请求

requestWithPatientGrant(doctor, patient, doctorSignature, patientSignature)对应的情况是医生在发起请求时就已经拿到了患者的许可, 这种情况下, 区块链除了记录下请求外, 还会立即通知医院为该请求生成索引, 请求成功后医生就可以直接通过mongoProxy拿到患者的数据了

患者可以同意或拒绝某一个请求, 这都是通过向区块链提交一笔新的交易来完成的
grant(patient, txid, doctor)表示患者同意医生某一个请求, 这种情况下, 区块链会记录下如下信息:

```
"grant" patient txid  ->  txTimestamp
```

并向医院发出通知, 让医院为该请求生成索引

目前暂时没有显式的拒绝, 因为 不显式同意 = 拒绝

FAQ
1. 如何确保医生的身份, 即保证所有的请求都是由合法的医生所发起的?

区块链首先需要记录下所有医生的信息, 最基本的信息包括医生的姓名和他的公钥, 除此之外, 区块链会记录下该医生的最后一个请求的请求号, 请求号不等于txid, 它只是一个单调递增的数字而已, 没有其它任何含义.
医生发起请求时, 请求参数为: (doctor, patient, sig)
其中, sig = sig(doctor, patient, N), sig()函数表示用医生的私钥进行签名, N是一个请求号, 医生需要保证每一个新的请求的请求号N都比上一个请求的请求号大即可
当区块链收到请求(doctor, patient, sig)时, 会用医生的公钥对sig部分进行解密, 通过判断N的递增性来验证医生的合法性

i. 如何确保病人的身份, 即所有的grant都是由合法的患者所发起的?
   同faq1, 区块链记录所有患者的信息, 最基本的应该包括患者的姓名和他的公钥

ii. 如何确保医院的身份?
   暂时不考虑这个问题, 现阶段医院不会主动向区块链提交任何交易请求