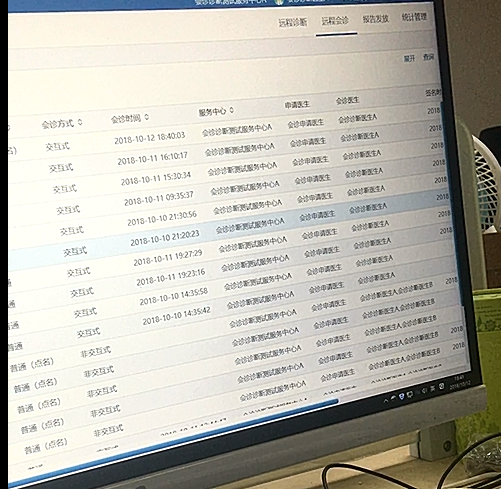
视频提取

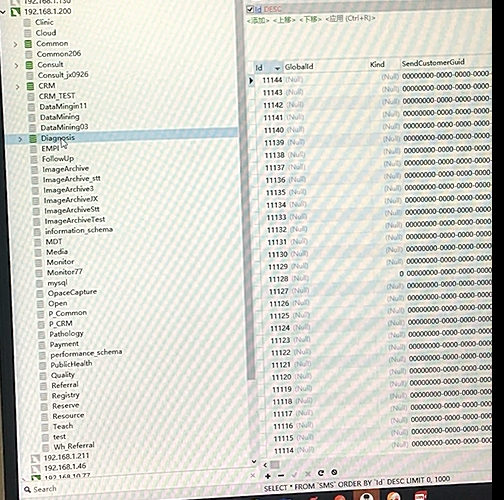
是

（1）

比如说会诊，我们首先要看医生和哪些服务中心有签约。把签约的数据查出来。



然后去会诊表里去找。我们一般只管这两个库——会诊和诊断。



要操作的两个数据库——192.168.1.200：

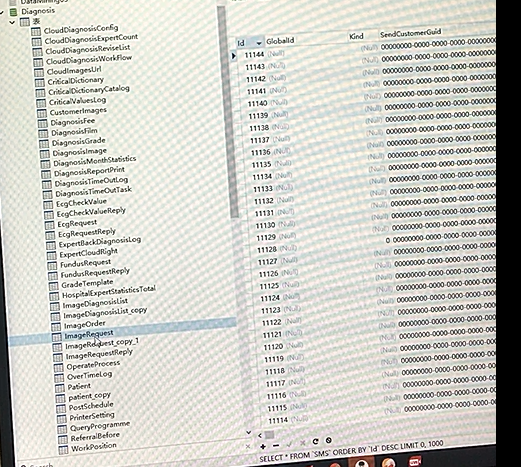
Consult（会诊）和Diagnosis（诊断）两个数据库。

Diagnosis（诊断）数据库中比较重要的三个表：

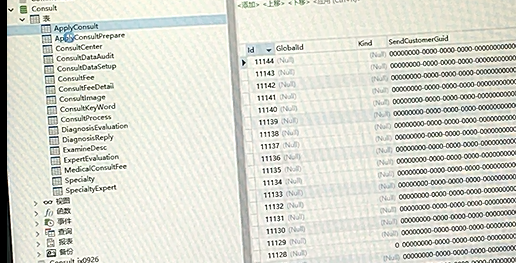
ImageRequest（影像诊断）

EcgRequest（心电诊断）

FundusRequest（眼底诊断）



Consult（会诊）数据库：

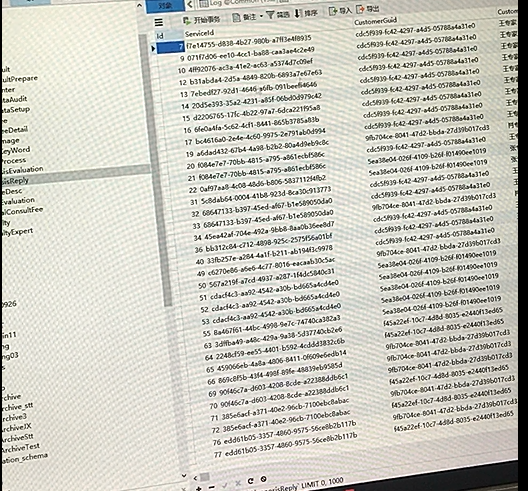


ApplyConsult：申请表。

一条会诊可能有多个医生，但每个医生只能查直接的，所以要去关联DiagnosisReply这张表。这张是医生的回复表。

前面写的医生的报告，就保存在这里。然后再加上用户的id，还有服务中心的id，各种连表，连接查询。

李阳：连是只连了两张，目前测试下来是还可以，速度。

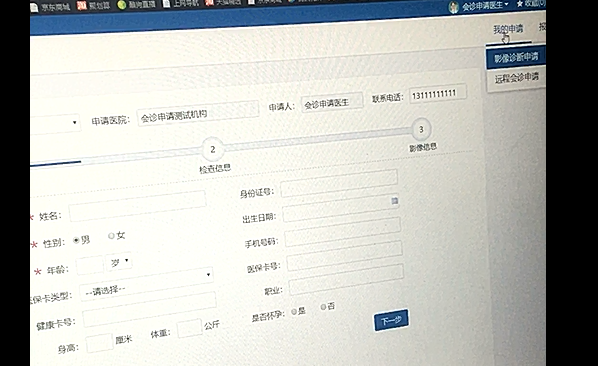


用连表去查询。会慢。所以把 我的任务 和 任务领取给单独分开了。一个是属于你的，而另一个，非交互式是需要领取的。（相当于抢单一样？）

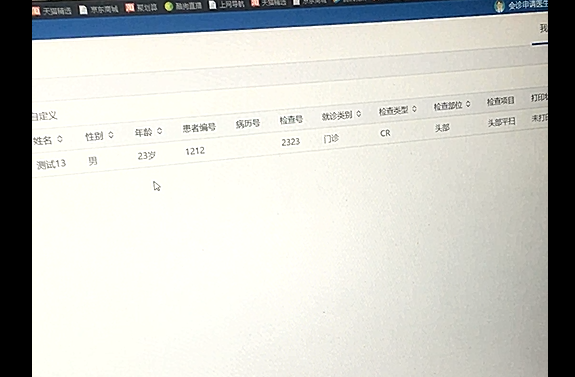


接下里是诊断，俞永红给我讲的。

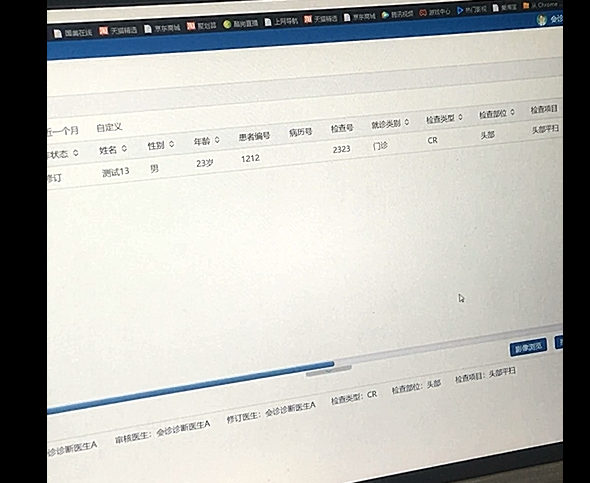
Web端就是医生申请，然后等着结果，（应该是上级的医生接单，然后给回复，我是这么想的）



申请完之后，这里就会有列表。

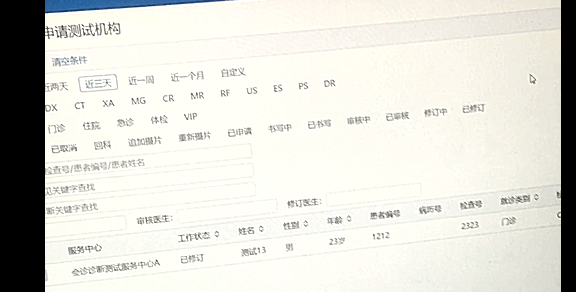


能看到其他医生处理完的列表。



根据条件去查询，俞永红表示，如果好几张表都会有关联的话，那查起来就慢，花的时间很长。我想，这个地方就要用的Lucene.net，或者说es了，每次去数据库里查，太不现实了。

反正现在的首要任务是把所有的都搞定，我也熟悉项目和流程，到时候根据需求再去做一个完美的版本！



申请端有一个统计：



他（也就是当前登陆的社区医生？）主要关心的是，他在某一个时间段，向哪些服务中心，申请了多少量。

就好比我们这家医院向你这个服务中心（省级大医院）提交了多少的量来。

俞永红又纠正，是我自己，也就是当前登陆的医生，他自己。他关心自己提交了多少的量，然后服务中心那边完成了多少。

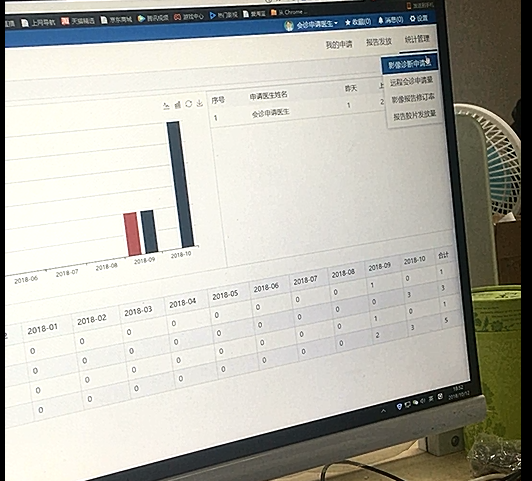
登陆医生所能申请的服务中心，必须是他所在的那家医生，和该服务中心又结对关系。

那该医生，就可以把申请给提交到结对的服务中心去。

然后服务中心的专家就可以把这些单子给做了。（我忽然想到，也不一定需要按工作流啊，因为它是非固定的，必须怎样流动的一条曲线。比方，我这边提交了申请，放入数据库，那服务中心那边的web客户端，只要有一个线程在轮询，一直去数据库里找数据不就可以了，然后显示在网页上，只要医生就可以抢任务了。我想到，这个轮询，也可以是定期地用ajax去查找）

我们的产品出去，肯定就是医院。这家医院肯定有这个能力，它买下这个产品，把区域内的卫生院都接进来。然后它开设一些服务中心出来，那小卫生院，比方是做ct的，那和该服务中心建立结对关系之后，ct都提交到服务中心这个平台，平台给你做好，那小卫生院只管拿报告。

申请，差不多就是一个申请，一个统计。



统计的话，统计他的申请量。服务统计的话，就相当于统计该医生申请上来的单子，被哪些专家给做了。然后做得怎么样。评判标准，做的快不快，然后做好之后，有没有进行多次的修改，如果多次修改了，肯定就做的一般了。如果修订率少的话，肯定说明这个专家比较专业。

患者是没有权利查看，而是依托于申请机构，它负责把你的病情往大城市的大医院去申请，替你发送一条影像诊断的申请，然后把结果反馈给你。



钱的话，他们自己去算。我们不管，我们只管服务。

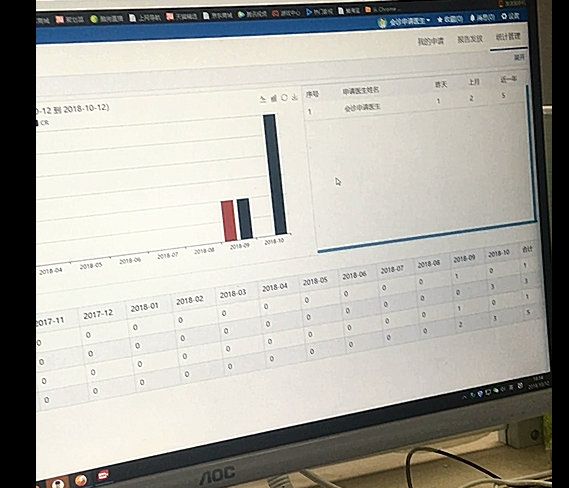
卫生院这里的统计，方便它和服务中心去结算。卫生院向服务中心支付相应的费用。它就按下面的表格，对每一家服务中心进行结算。

然后最后服务中心又会有一个统计，它去给那些专家进行结算。（做了事，总不能不给钱吧。对吧。）

俞永红就是做这个申请和统计。原来有其他同事做过，但都走了。以后都是我们几个负责了。

老版本的问题就是，我们修改了新版本，我们觉得用不到的东西，给删了。但医院不肯，说是我们改坏了。

这个项目已经上线，已经有医院在用了。



（2）