群里的消息整理

1.

研发-姜良敏(499367160) 14:18:08

from c in \_customerRepository.Table from r in c.CustomerRoles where r.Id == customerRolesId select c;

query.Where(p => p.CustomerRoles.Any(s => s.Id == customerRolesId));

研发-姜良敏(499367160) 14:19:09

之前的根据角色查询用户信息：

之前的 query.Where(p => p.CustomerRoles.Any(s => s.Id == customerRolesId));

研发-姜良敏(499367160) 14:19:22

现在需要改成：from c in \_customerRepository.Table from r in c.CustomerRoles where r.Id == customerRolesId select c;

研发-姜良敏(499367160) 14:19:27

关联

产品-孔钧(6936816) 14:22:56

之前的角色的写法产生的SQL性能有很大的问题，Any的数据正确性也存疑，请大家都按照姜良敏发的改一下，下次代码抽查希望不要再发现这样的写法

@产品-简刚 @研发-李鹏飞 @hk @研发-曹兵 @研发-冯鸣亮 @研发-王健 @研发-袁法学 @研发-张军 @！！！ @研发-李彪 @研发-李图斌 @研发-姜良敏 @研发-胡静杨

产品-孔钧(6936816) 14:23:58

将 query.Where(p => p.CustomerRoles.Any(s => s.Id == customerRolesId)); 改为 from c in \_customerRepository.Table from r in c.CustomerRoles where r.Id == customerRolesId select c;

研发-姜良敏(499367160) 14:37:42

这边300万数据

研发-姜良敏(499367160) 14:37:49

Where(p => p.CustomerRoles.Any(s => s.Id == customerRolesId))

研发-姜良敏(499367160) 14:37:55

这个是不行的

研发-姜良敏(499367160) 14:37:58

CustomerRoles.Any

研发-姜良敏(499367160) 14:38:04

from c in \_customerRepository.Table from r in c.CustomerRoles where r.Id == customerRolesId select c;

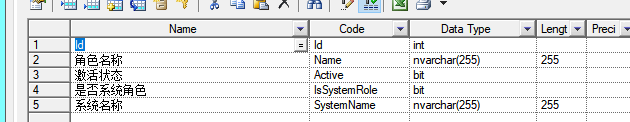
研发-姜良敏(499367160) 14:38:14

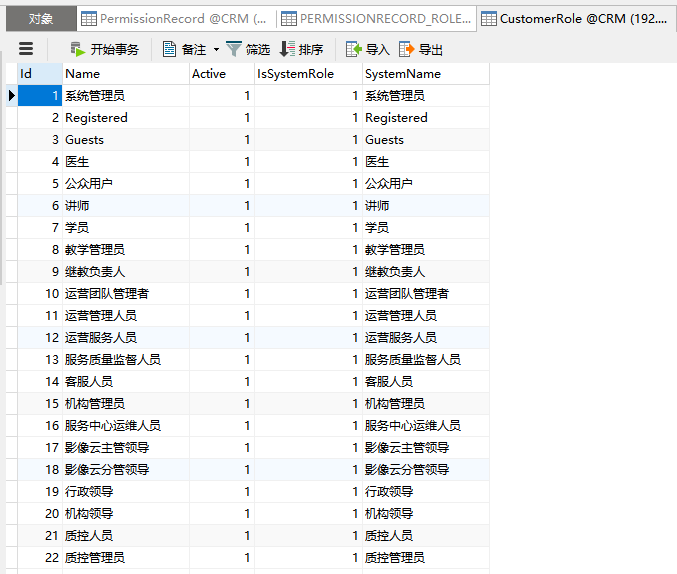
关联的秒出

顺便查找角色，权限关系。

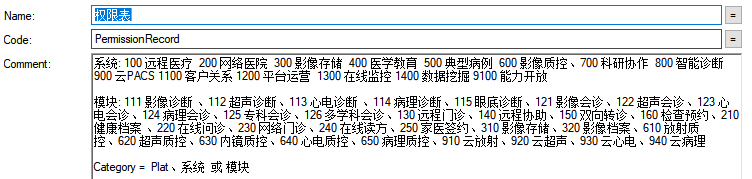
（1）

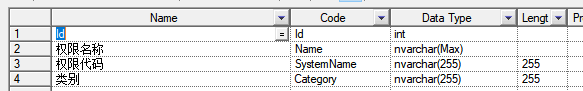
角色表：

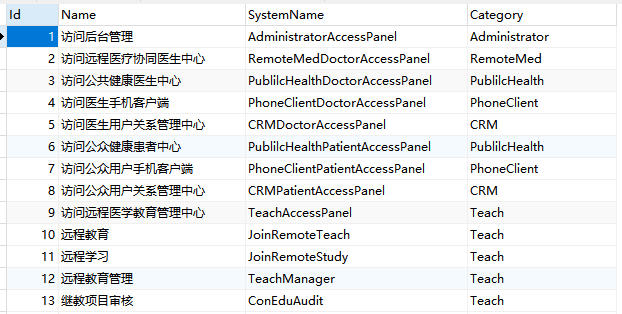




（2）权限表：



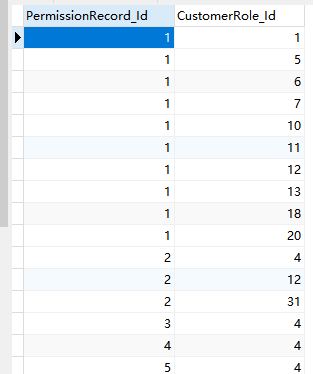




（3）权限角色关联表

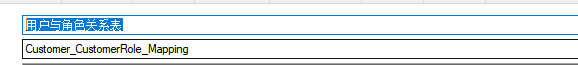




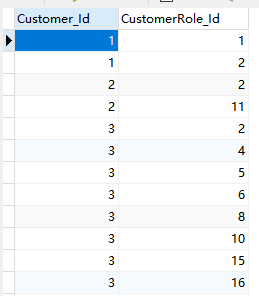


比方这里的1，对应权限——访问后台管理，这里可以访问的角色就是，1（系统管理员），5（公众用户），6（讲师），7（学员），10(运营团队管理者)，等等。

（4）用户与角色关系表







这个到底有没有在用，我还真不知道。

从代码上看，

from c in \_customerRepository.Table

from r in c.CustomerRoles

where r.Id == customerRolesId

select c;

这个代码看，要么customer表中有一个CustomerRoles的字段，但是找了下，好像没有，那应该就是一对多，存在这么一个关系啊，但应该有中间表才对啊，暂时没有找到。

找到了，在map中是存在着多对多关系的：

public partial class CustomerMap : EntityTypeConfiguration<Customer>

{

public CustomerMap()

{

this.Ignore(u => u.PasswordFormat);

this.HasMany(c => c.CustomerRoles)

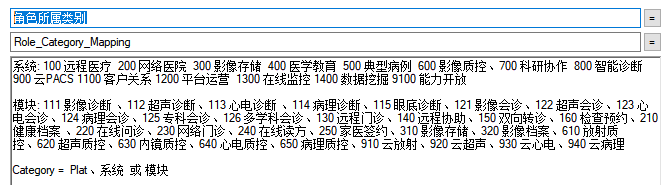
.WithMany()

.Map(m => m.ToTable("CUSTOMER\_CUSTOMERROLE\_MAPPING"));

}

}

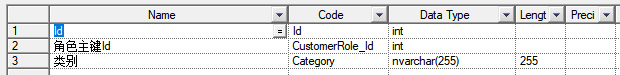
（5）角色所属类别

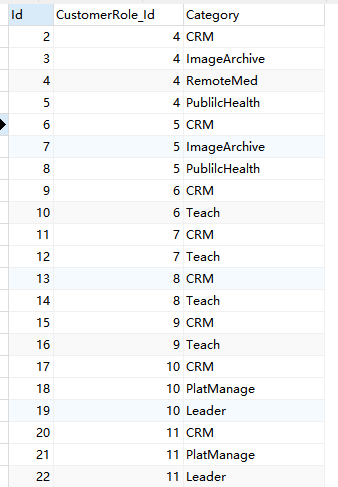


系统: 100 远程医疗 200 网络医院 300 影像存储 400 医学教育 500 典型病例 600 影像质控、700 科研协作 800 智能诊断 900 云PACS 1100 客户关系 1200 平台运营 1300 在线监控 1400 数据挖掘 9100 能力开放

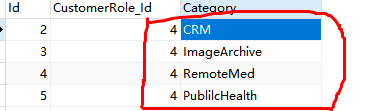
模块: 111 影像诊断 、112 超声诊断、113 心电诊断 、114 病理诊断、115 眼底诊断、121 影像会诊、122 超声会诊、123 心电会诊、124 病理会诊、125 专科会诊、126 多学科会诊、130 远程门诊、140 远程协助、150 双向转诊、160 检查预约、210 健康档案 、220 在线问诊、230 网络门诊、240 在线读方、250 家医签约、310 影像存储、320 影像档案、610 放射质控、620 超声质控、630 内镜质控、640 心电质控、650 病理质控、910 云放射、920 云超声、930 云心电、940 云病理

Category = Plat、系统 或 模块

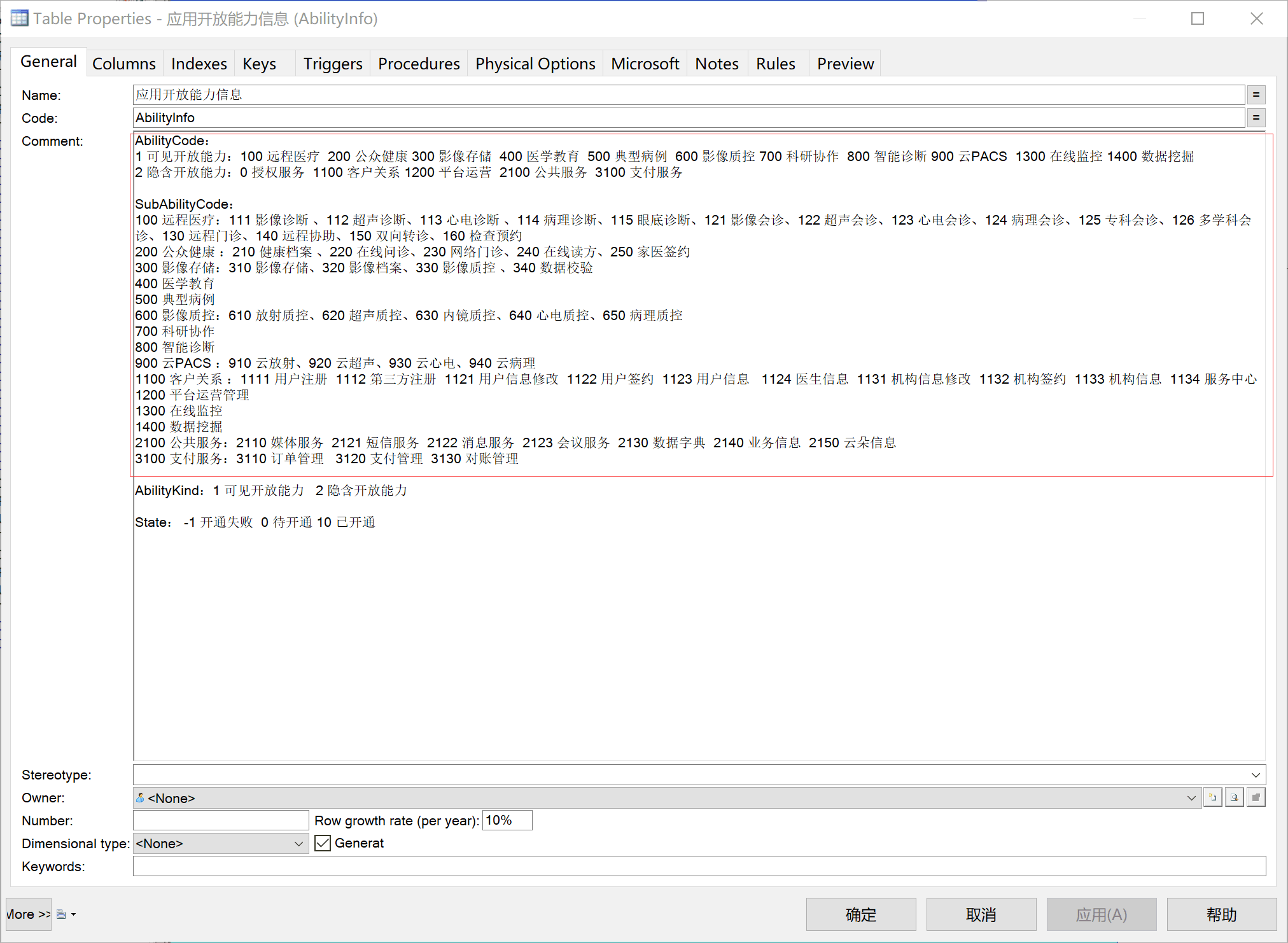




这里可以看出当前角色是适用于哪个子系统（我自己的理解），比方这里的角色4，也就是医生，它对应在4个子系统中——

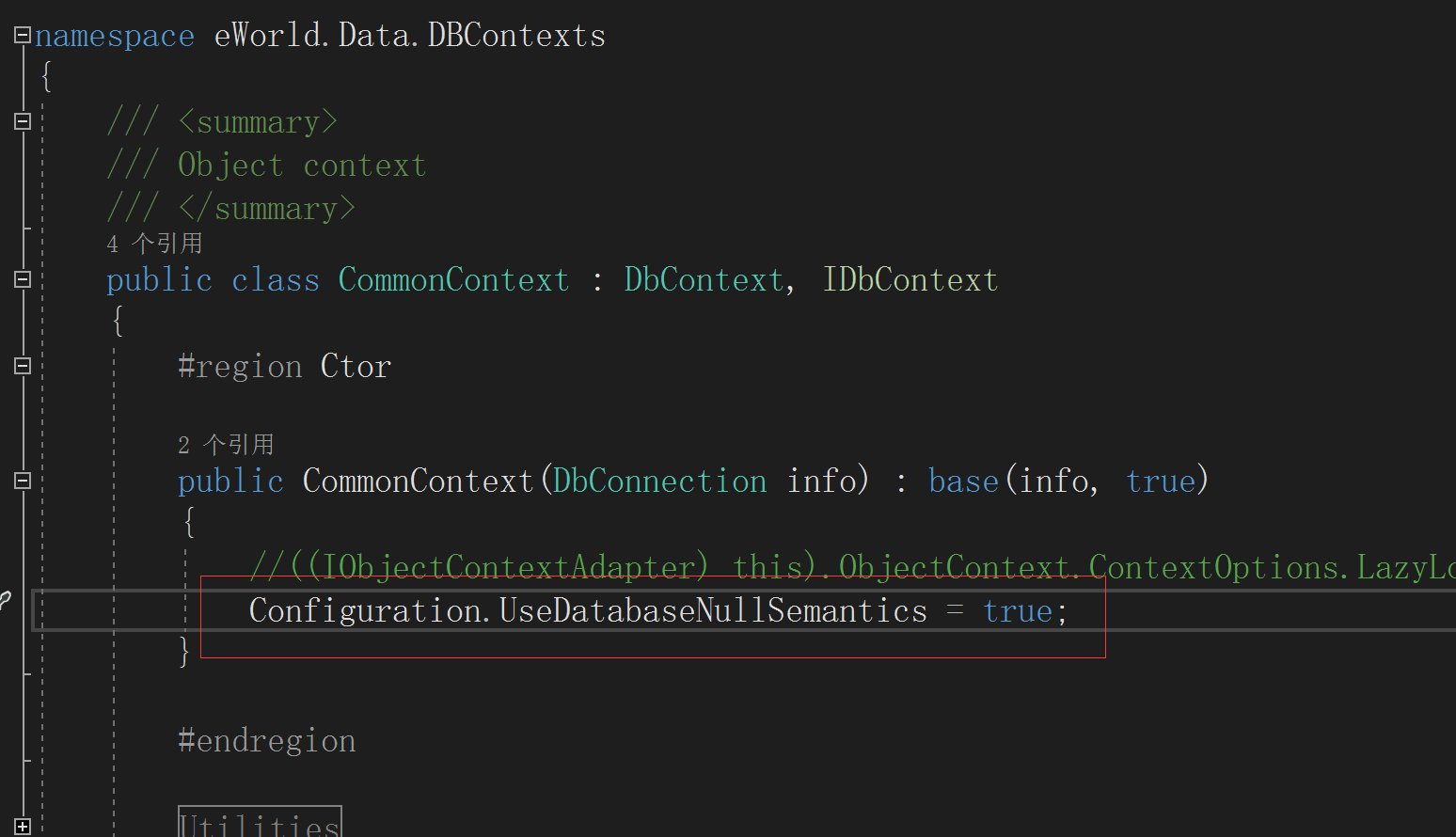


2.



@测试-沈婷婷 @研发-李彪 @hk 这块改成和授权一致了

3.



产品-孔钧(6936816) 16:23:03

这个语句大家还没加啊

产品-孔钧(6936816) 16:23:17

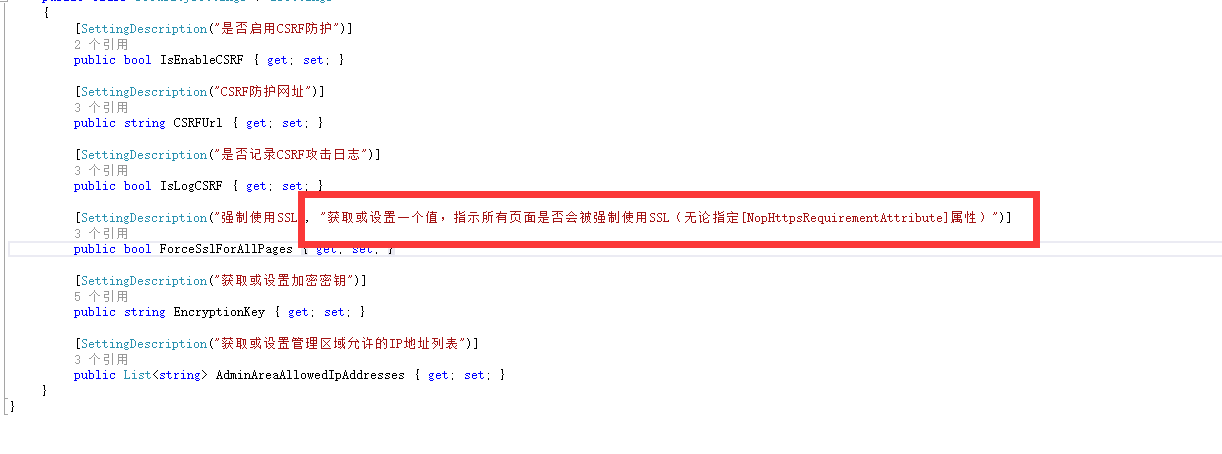
@研发-张军 @研发-袁法学 @研发-李彪 @hk @！！！ @研发-姜良敏

产品-孔钧(6936816) 16:24:52

利用值查询在WHERE条件上没有过多的条件过滤，而利用参数查询则是生成过多的条件筛选，到这里是不是就到此为止了呢，如果您对于参数查询不想生成对空值的过滤，我们在上下文构造函数中可关闭这种所谓【语义可空】判断

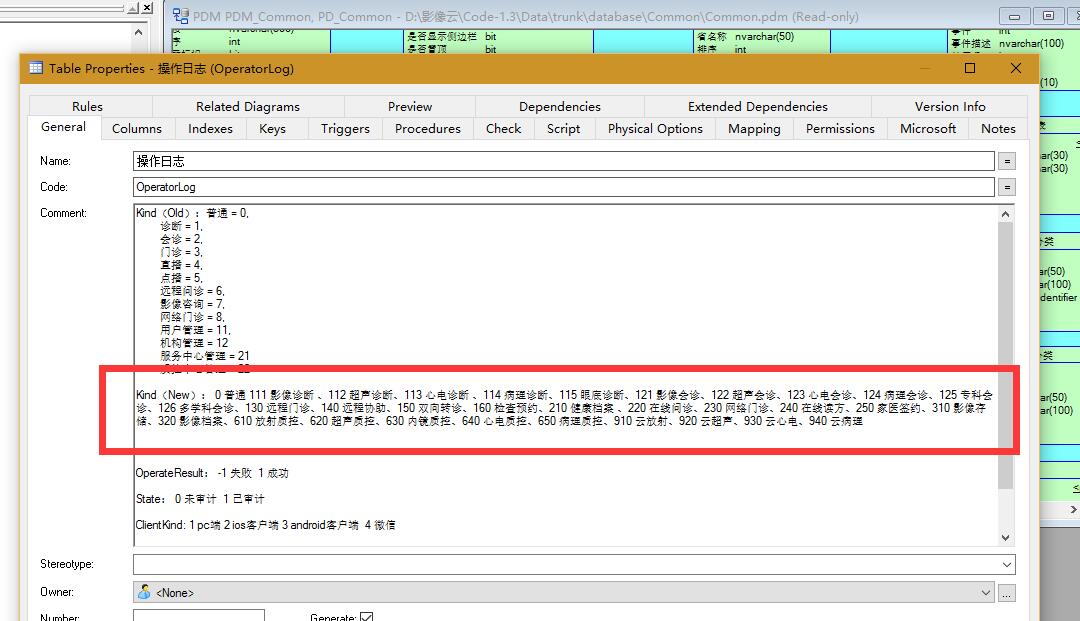
4.

关于Settings参数设置，现在很多参数都没加上备注说



请没加的都加上不然运营不知道什么意思什么用途《

5.



研发-姜良敏(499367160) 17:35:23

操作日志的的Kind要按照新的来

6.

角色和权限：

产品-孔钧 2019/2/26 16:09:45

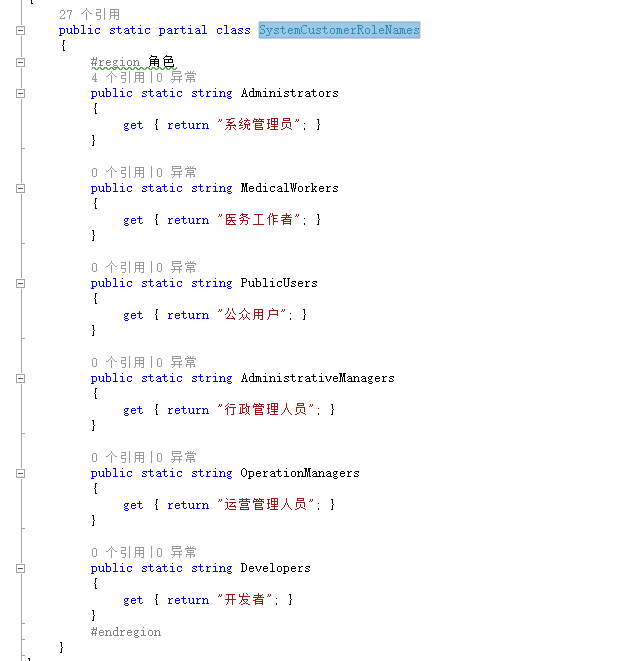
@全体成员 医网云角色和权限数据库设计、entity、角色和权限cs文件已更新，有问题直接来找我

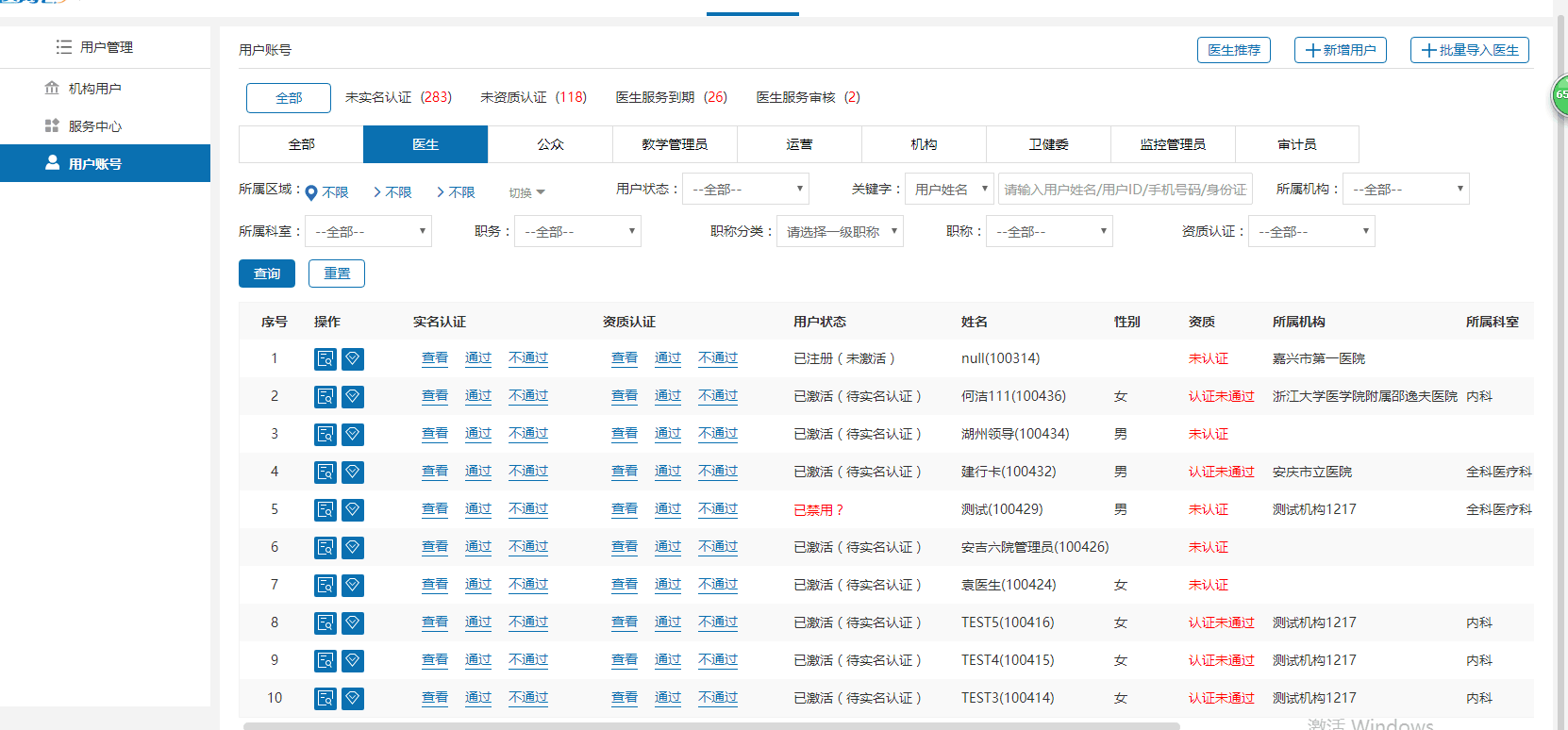
产品-孔钧 2019/2/26 16:11:05

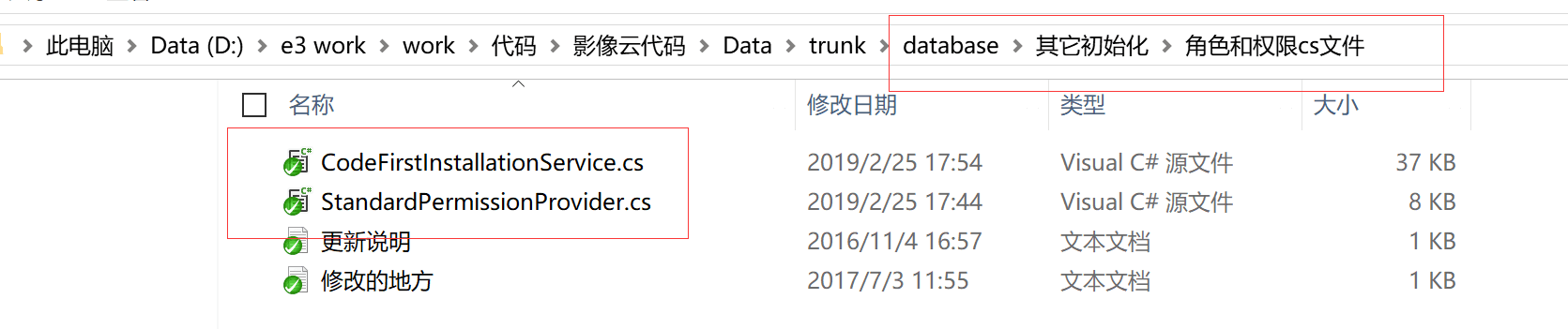
@研发-张军 @研发-姜良敏 权限组的功能已经可以开始开发

产品-孔钧 2019/2/26 16:11:47

@研发-袁法学 初始化程序可以获取重新更新了

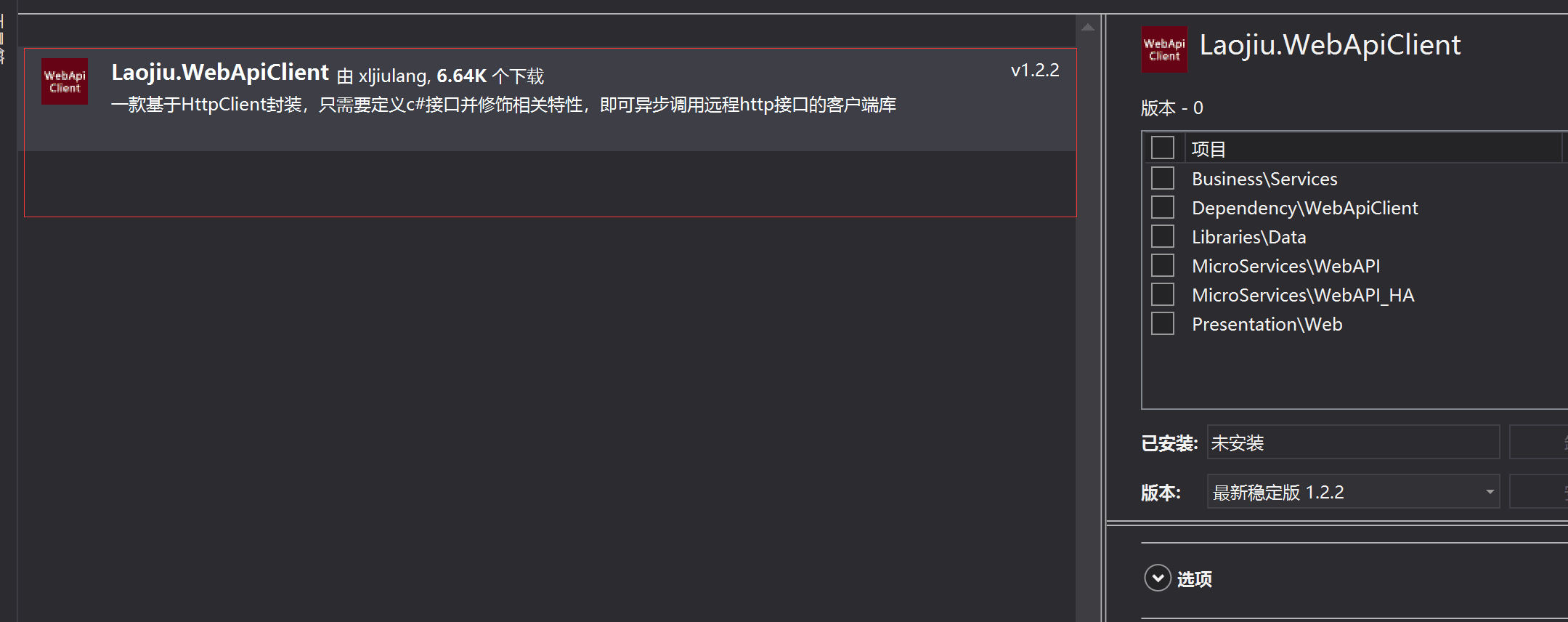






7.

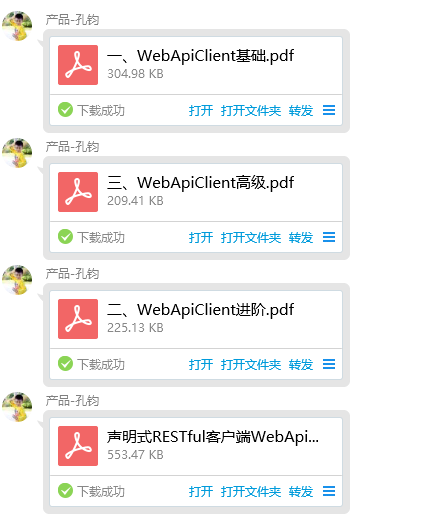
WebApiClient



目前先实现WebAPI的调用，后续跟微服务（SpringCloud的Steeltor）的集成我会专门进行培训的

产品-孔钧 2019/2/27 12:42:08

Steeltoe



8.

9.

10.