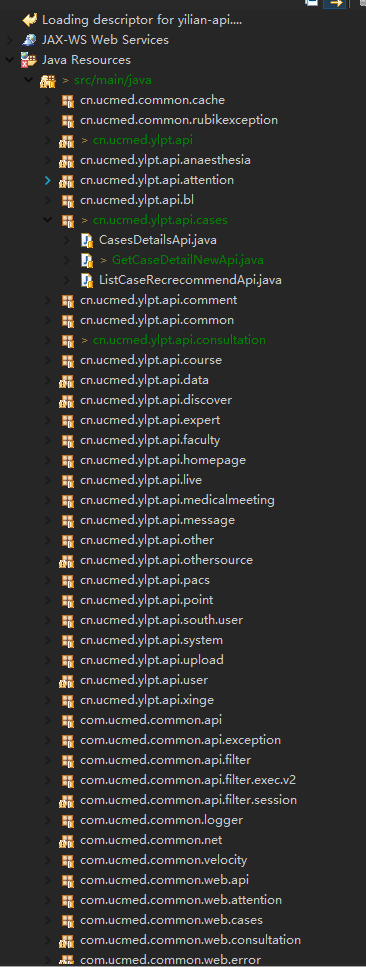
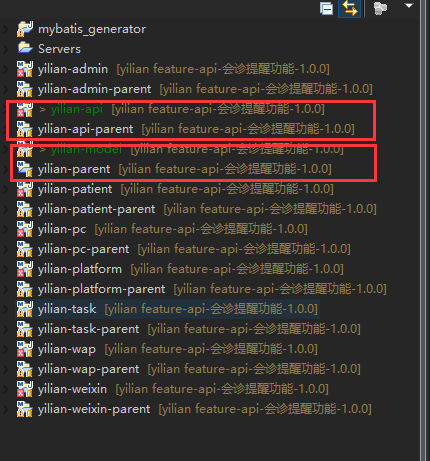
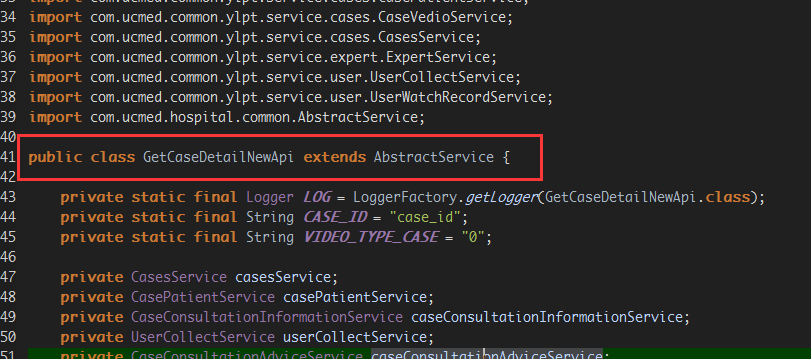
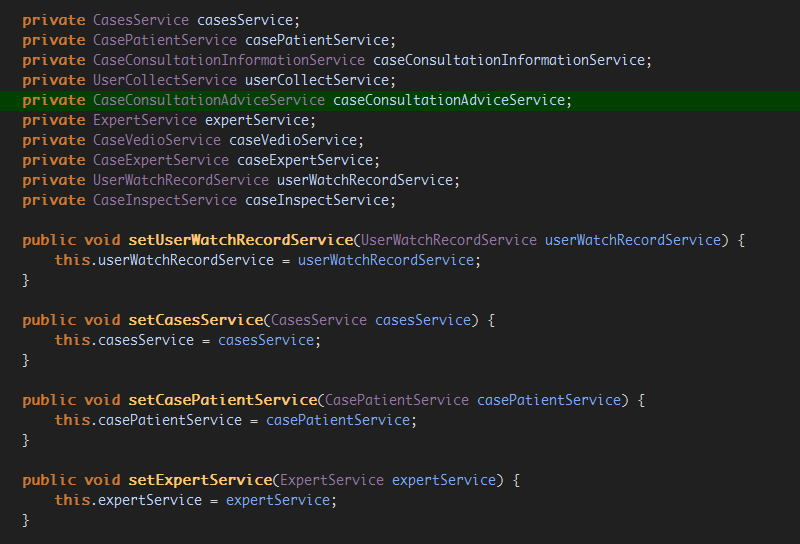
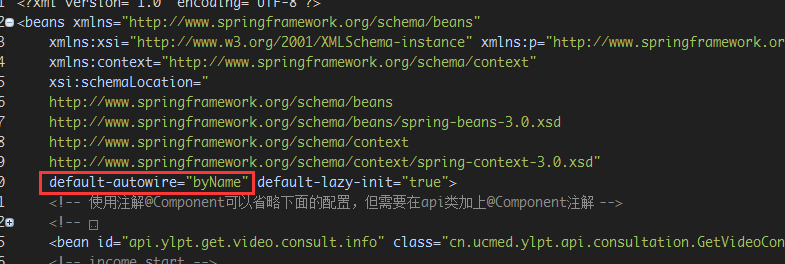
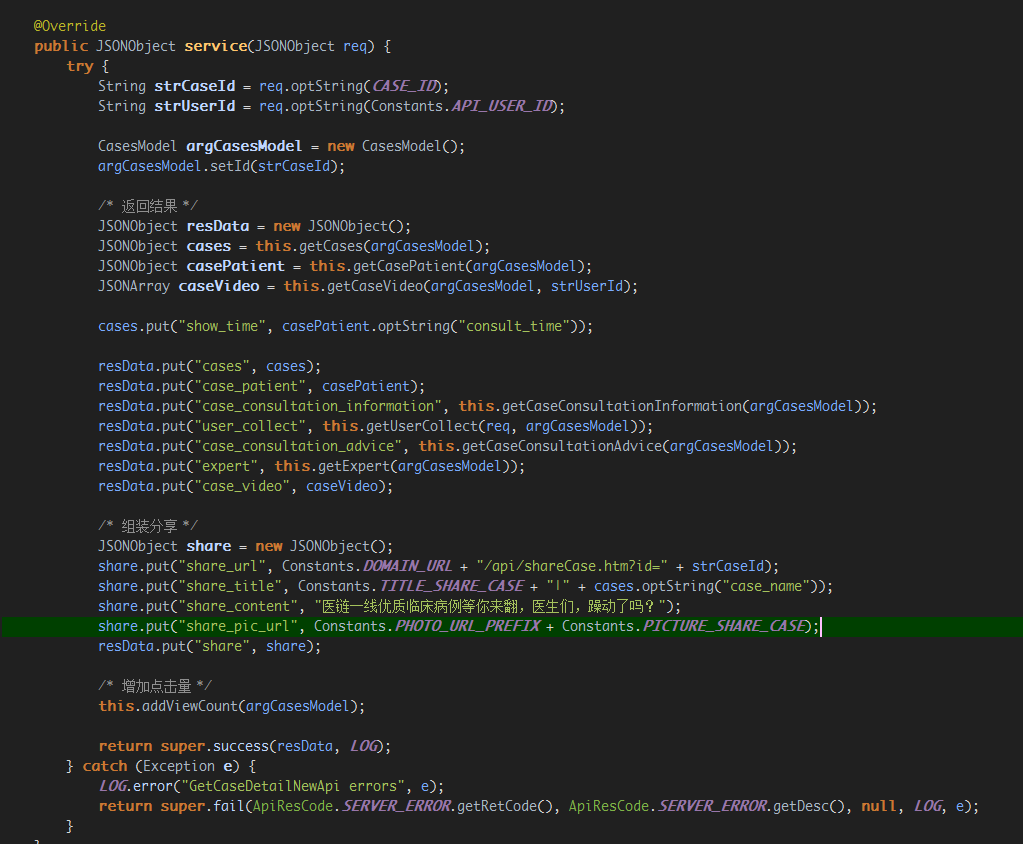
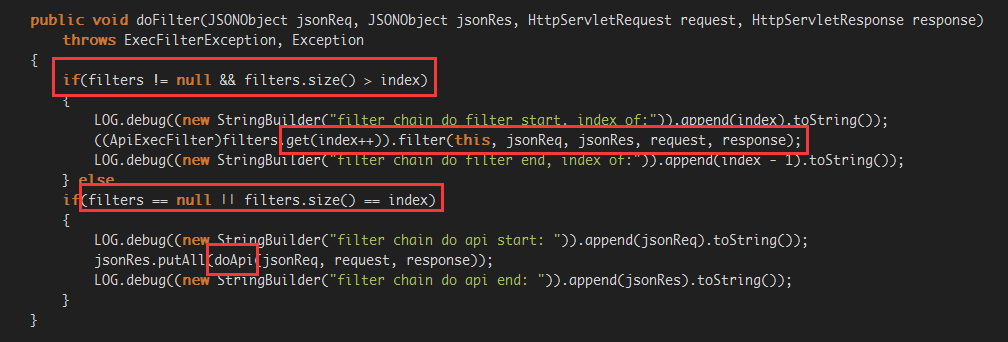
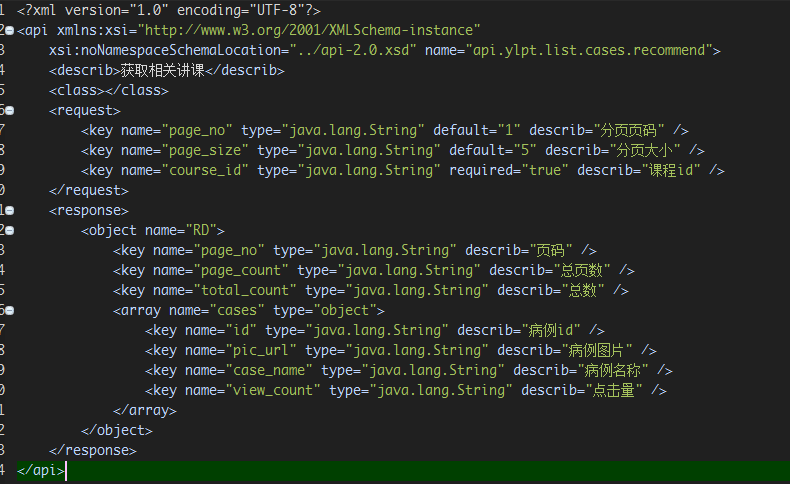
医链项目技术解析

@author Thomas

@version 2017/01/11 09:49

1. 架构
   1. Nginx + tomcat 负载均衡 + Web Service + 缓存（Memcached）
2. 用到的技术
   1. 技术框架
      1. Spring + Spring MVC + MyBatis
   2. 其他技术
      1. 服务器：Tomcat + Ngnix、负载均衡
      2. Web Service
      3. 缓存（Memcached）
      4. 日志（SLF4J）
      5. 富文本（Fckeditor）
      6. 推送（信鸽）
      7. 短信（玄武）
      8. 对象存储（COS）、图片上传（万象优图）、文件上传（单张或者多张）
      9. JDBC
      10. HttpComponents（主要提供对HTTP服务器的访问）
      11. 加密技术
      12. PDF转成H5、PPT转成图片
      13. 支付
      14. Maven
      15. Git、SVN
      16. 前端知识：html、css、js、json、ajax等
      17. SOAP
      18. 过滤器、拦截器
      19. 验证码
      20. Session和Cookie
3. 项目代码结构
   1. 
   2. 
      1. Yilian-api是用来放置业务逻辑层
      2. Yilian-model是用来放置业务层、持久层、对象（和其变型，VO和DTO）
   3. 例如
      1. Com.ucmed.ylpt.api.user
      2. Com.ucmed.common.ylpt.service.user
      3. Com.ucmed.common.ylpt.dataobject.user
      4. Com.ucmed.common.ylpt.model.user
      5. com.ucmed.common.ylpt.dao.user
4. 具体代码解析
   1. 
   2. 继承AbstractService，这样GetCaseDetailNewApi就可以直接执行对应的方法service();
   3. 
   4. 
   5. set方法注入，因为Spring Bean配置文件已经说明了GetCaseDetailNewApi其中的Bean的注入方式（注意，这里default-autowire=”byName”指的是被当前xml文件注入的Spring Bean内部注入其他Spring Bean的方式是“byName”，然后在GetCaseDetailNewApi中通过set方法注入即可）
   6. 
   7. 业务逻辑主要内容
   8. 具体的业务处理结果，需要进行字段转换，而且要进行相应的封装
   9. 不能分离的业务一定要进行事务管理，也就是说要写在service中
   10. 以上就是业务逻辑处理的截图，其他业务层和持久层以及领域对象层就不再进行赘述
   11. ================下面介绍框架流程======================
   12. 
   13. 
   14. 
   15. 基本就是通过Spring的帮助，先执行filter，然后在（filter == null || filter.size() == index）的时候处理相应的xxx.service()（eg. GetCaseDetailNewApi.service()），也就是doApi(jsonReq, request, response)，期间需要校验按照SOAP规范的请求XML和响应XML格式。（具体内容可以通过JadClipse进行反编译查看源代码）
5. 注意事项