哈姆瑞特项目分析

主题

分析功能点

引导页 —cms获取分类下的文章即可

交流区

1寻找目标专家医生-私聊或者群聊

2.修改我的病例

活动区

发现区

1.工作人员加入和管理病例库

个人区

1.个人信息查看接口

2.个人信息修改接口

3.增加我的随访，列出和上传患者随访资料，医生可列出和上传患者的随访资料。在“我”区增加“我的随访”功能模块， 添加随访方案模板（按疾病病种分类，有房颤、室上速、房扑、室早、室速、房速、起搏器、左心耳封堵、其他，列表式）

* 分析主体

病例管理：

属性：病例ID 创建者ID 患者ID 诊断信息 病情信息 是否归档到病例库 备注 备用1 备用2 备用3 备用4 备用5

动作：增删改查

病例标签管理：

属性： 标签id 创建者用户ID 标签名称 创建标签日期

动作：增删改查

病例标签关系表：

属性：表ID 病例ID 标签ID

患者信息：跟随病例创建和删除

属性：患者ID 创建者用户ID （基本信息）姓名 性别 出生日期 年龄 手机号 身份证号 —（患者信息）就诊日期 科室 （编号信息1）编号类型 编号 （编号信息2） 编号类型 编号 （更多）固定电话 邮箱 患者地址 患者职业 患者联系人 介绍人 备注 等

动作：

查看个人信息—待定

医生 ： 普通医生 专家医生 涉及到医生级别A-预留 B-预留 C-普通医生 D-预留 E-专家医生 F-预留

属性：医生等级 年龄 性别 是否接收转诊 积分 等

动作：

发起转诊

查看聊天清单

转诊群聊

辅助便捷操作

聊天设置

管理员：对所有信息可以控制

操作员 ：工作人员后台配置展示信息比如资料库和病例库，病例库是在每次转诊交流中有医生们逐步创建的，是系统的核心资产，需进行有效的管理，以备后期扩展功能功能

转诊群组信息：

表ID 发起人ID 群组名称 创建时间 删除时间

群组成员表

表ID 群组ID 成员ID

转诊群组聊天信息：如果聊天过程发送了病例则保存相应的病历号ID

表ID 群组ID 用户ID 内容有病历号的病历号ID 普通内容 时间

* 分析业务流程—转诊
  + - 1.A医生-我的信息-发布和修改病例
    - 2.A医生浏览病例列表和病例信息
    - 3A医生寻找目标医生专家B（接收转诊的专家）
    - 4A和B私聊 或者 ABC群聊》保存聊天信息

5.医生发起转诊，上传资料，选择目标医生，确认转诊

6.B专家需要设置是否接收转诊

* 分析权限
* 分析技术实现
  + 1.仅做APP，后台提供CMS和接口服务
  + 2.接口服务区分普通接口 和 登录验证接口
  + 3.普通接口直接获取数据库信息即可
  + 4.登录权限校验使用 token+key方式

改造方案：

1.用户表扩展一个字段标示《医生用户》，检索医生使用。

还需要扩展用户坐标信息x,y，检索附近的医生使用-扩展考虑

用户角色中加入 医生角色 如果医生分级别，可以考虑分多个医生角色管理

2.引导页直接用接口获取普通文章信息，后台管理可以修改添加。

4.病例 新建表-病例表—-或者创建一个病例模块维护此类别，看实际具体需求，如果创建表则需要

id 文章Id（看情况） 简称 全称 描述 时间 创建人id 创建时间 最后修改时间 最后修改人

5 病例修改历史记录表—-看情况

id 病例id 修改人 修改时间 修改内容 备注

6.病例库，在后台管理新建文章分类。

——————-表设计

图片上传 hurt\_upload

id varchar(64) 表id

userid varchar(64) 用户id

caseid varchar(64) 关联其他id 例如病历上传 关联articleid

remark varchar(512) 图片备注

path varchar(512) 图片链接地址

活动点赞 :hmrt\_like

id varchar(64) 表id

userid varchar(64) 点赞用户id

articleid varchar(64) 点赞文章id

患者表：hmrt\_patient

患者ID 创建者用户ID （基本信息）姓名 性别 出生日期 年龄 手机号 身份证号 —（患者信息）就诊日期 科室 （编号信息1）编号类型 编号 （编号信息2） 编号类型 编号 （更多）固定电话 邮箱 患者地址 患者职业 患者联系人 介绍人 备注 创建日期 等

id varchar(64) 患者ID

userid varchar(64) 创建者用户ID

name varchar(128) 姓名

sex char(1) 0男 1 女 性别

birthday datetime 出生日期

age int 年龄

mobile varchar(32) 手机号

id\_number varchar(32) 身份证号

visit\_dt datetime 就诊日期

department varchar(128) 科室

sn\_type1 varchar(64) 编号类型

sn\_no1 varchar(64) 编号

sn\_type2 varchar(64) 编号类型

sn\_no2 varchar(64) 编号

phone varchar(64) 电话

email varchar(128) 邮箱

address varchar(512) 地址

profession varchar(64) 职业

contact varchar(64) 联系人

introduce varchar(64) 介绍人

remark varchar(512) 备注

create\_dt datetime 创建日期

—病历表=文章表

加入字段

—-已有articleid 病例ID

—-已有userid 创建者ID

patientid varchar(64) 患者ID

diagnose\_info varchar(512) 诊断信息

condition\_info varchar(512) 病情信息

isarchive char(1) 0否 1是 是否归档到病例库 默认1

—医生信息表 = 人员表

level char(1) 医生等级 医生级别A-预留 B-预留 C-普通医生 D-预留 E-专家医生 F-预留

hospital varchar(512) 医院名称

isturn char(1) 是否接收转诊

score varchar(64) 积分

token varchar(64)

openId varchar(64) 第三方登录id

——-病例标签 hmrt\_tags

id varchar(64)

userid varchar(64) 创建者用户ID

name varchar(128) 标签名称

create\_dt datetime 创建标签日期

———-病例标签关系表 hmrt\_article\_tags

id varchar(64)

articleid varchar(64) 病例ID

tagsid varchar(128) 标签id

————— 转诊群组信息：hmrt\_group

id varchar(64) 表ID

userid varchar(64) 发起人ID

name varchar（128） 群组名称

create\_dt datetime 创建时间

dismiss\_dt datetime 删除时间

—-群组成员表 hmrt\_group\_user

id varchar(64) 表id

userid varchar(64) 成员ID

groupid varchar(64) 群组id

—————— 转诊群组聊天信息：hmrt\_group\_chat

如果聊天过程发送了病例则保存相应的病历号ID

id varchar(64) 表ID

groupid varchar(64) 群组ID

userid varchar(64)用户ID

articleid varchar(64) 聊天过程中发送病例—保存病历号ID

content varchar(512) 普通内容

create\_dt datetime 时间

———-接口———————

| 接口说明 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 哈姆瑞特接口说明  1、接口访问地址：<http://localhost:8181/jeesite/>f/ ，具体业务接口可以拼接URL。  2、输入输出参数说明仅供参考  3、代码位置参考URL一级地址  4、接口统一数据校验规则如下：  String key = StringUtil.toString(request.getParameter(“key")); 公共秘钥  String sign = StringUtil.toString(request.getParameter(“sign")); 签名  String t = StringUtil.toString(request.getParameter(“t")); 时间  通过私有秘钥进行验签：Constants.SECRET\_KEY  String md5 = Md5.encrypt(key + t + Constants.SECRET\_KEY); 验签规则  所有接口都要进行校验 md5 和 sign 进行比对  5、除了此接口校验，其余需要登录查询的信息还需要传入公共参数：  userid 用户ID  token 登录返回的临时用户校验令牌 | | | | | | |
| 模块 | 子模块 | **URL** | 输入 | 说明 | 输出 | 说明 |
| 引导页 | 轮播图 | mainApi/scrollImages | 无 |  | id |  |
|  |  | title |  |
|  |  | content |  |
| 用户信息 | 登录 | userApi/login |  |  |  |  |
|  | 注册 | userApi/reg |  |  |  |  |
|  | 查询个人信息接口 | userApi/queryUserInfo |  |  |  |  |
|  | 重置密码接口 | userApi/resetPwd |  |  |  |  |
|  | 修改个人信息接口 | userApi/updateUserInfo |  |  |  |  |
|  | 查询目标医生接口 | userApi/queryTargetDoctor |  |  |  |  |
| 标签信息 | 保存病例标签接口 | tagsApi/save |  |  |  |  |
|  | 查询用户所有病例标签接口 | tagsApi/queryUserTags |  |  |  |  |
|  | 删除用户病例标签接口 | tagsApi/deleteUserTag |  |  |  |  |
| 患者信息 | 保存患者信息 | patientApi/save |  |  |  |  |
|  | 查询患者信息 | patientApi/queryById |  |  |  |  |
| 病例信息 | 创建病例 | caseApi/save |  |  |  |  |
|  | 修改病例 | caseApi/updateById |  |  |  |  |
|  | 删除病例 | caseApi/deleteById |  |  |  |  |
|  | 查询用户病例列表 | caseApi/queryListByUserid |  |  |  |  |
|  | 查询用户病例详情 | caseApi/queryById |  |  |  |  |
| 病例库 | 查询病例库列表 | caseApi/queryArchiveList |  |  |  |  |
|  | 查询病例库病例详情 | queryArchiveById |  |  |  |  |
| 资料库 | 查询资料库列表 | infoRepoApi/queryInfoRepoList |  |  |  |  |
|  | 查询病例库病例详情 | infoRepoApi/queryInfoRepoById |  |  |  |  |
| 活动 | 活动列表查询 | activityApi/queryActivityList |  |  |  |  |
|  | 活动详情查询 | activityApi/queryActivityById |  |  |  |  |
|  | 活动点赞 | activityApi/activityLike |  |  |  |  |
|  | 活动点赞取消 | activityApi/activityLikeCancel |  |  |  |  |
|  | 活动点赞查询 | activityApi/activityLikeList |  |  |  |  |
|  | 活动评分+评论保存 | activityApi/saveComment |  |  |  |  |
|  | 活动评分+评论查询 | activityApi/queryCommentList |  |  |  |  |
| 检索 | 根据关键字检索资料库病例库用户 | searchApi/queryInfoByKeyWord |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

—————————————————

连接demo：

http://localhost:8181/jeesite/f/userApi/queryTargetDoctor

http://localhost:8181/jeesite/f/infoRepoApi/queryInfoRepoList

引导页 mainApi

轮播图接口 - 获取引导页的图片地址 超链接 等信息 scrollImages

用户信息 -userApi

登录接口 login

注册接口 reg

查询个人信息接口 queryUserInfo

重置密码接口 resetPwd

修改个人信息接口 updateUserInfo

查询目标医生接口 queryTargetDoctor

标签信息- tagsApi

保存病例标签接口 save

查询用户所有病例标签接口 queryUserTags

删除用户病例标签接口 deleteUserTag

患者信息 patientApi

保存患者信息 save

查询患者信息 queryById

病例信息 caseApi

创建病例接口 save

修改病例接口 updateById

删除病例接口 deleteById

查询用户病例列表接口 queryListByUserid

查询用户病例详情接口 queryById

病例库- caseApi

查询病例库列表信息 queryArchiveList

查询病例库病例详情信息 queryArchiveById

资料库 - infoRepoApi

查询资料库列表信息 queryInfoRepoList

查询资料库资料详情信息 queryInfoRepoById

活动 - activityApi

活动列表查询接口 - queryActivityList

活动详情查询接口 -queryActivityById

活动点赞接口 - activityLike

活动点赞取消接口 - activityLikeCancel

活动点赞查询接口 - activityLikeList

活动评分+评论保存接口 - saveComment ——-

活动评分+评论列表查询接口 - queryCommentList —-

检索—-searchApi

根据关键字索索病例库和资料库和用户 名称 —-queryInfoByKeyWord

聊天信息 - 待定

创建群组接口

保存群组用户接口

修改群组用户接口

保存群组聊天信息接口

保存聊天信息中包含病例信息的病例ID

好友 - 待定

添加好友接口

删除好友接口

通知 -待定

通知列表接口-待定

通知详情接口 -待定

————————————————-备注————————————————

protected int validate(HttpServletRequest request) throws Exception {

if (StringUtil.isEmpty(request.getParameter("key")) || StringUtil.isEmpty(request.getParameter("sign")) || StringUtil.isEmpty(request.getParameter("t"))) {

return -1;

}

String key = StringUtil.toString(request.getParameter("key"));

String sign = StringUtil.toString(request.getParameter("sign"));

String t = StringUtil.toString(request.getParameter("t"));

String md5 = Md5.encrypt(key + t + Constants.SECRET\_KEY);

if (!md5.equals(sign)) {

return 0;

}

return 1;

}

if (StringUtil.isEmpty(request.getParameter("userToken"))) {

outputJson(response, JsonUtil.beanToJson(putResponseData(401, "请求参数错误,userToken不能为空！", "")));

return;

}

int userToken = StringUtil.toInt(request.getParameter("userToken"));

String strSQL = " where token = " + userToken;

Member member = memberService.getMember(strSQL);

if(member == null){

outputJson(response, JsonUtil.beanToJson(putResponseData(501, "userToken错误,请重新登录", "")));

return;

}

Date date = new Date();

Integer token = Math.abs((int)date.getTime());

super.modifyMemberCondition("token = " + StringUtil.toString(token) + " where id = " + memberId);

String salt = Md5.encrypt(member.getMobile());

String md5newPassword = Md5.encrypt(salt + newPassword + "baidai");

strSQL = " salt = '" + salt + "', password = '" + md5newPassword + "' where id = " + StringUtil.toString(member.getId());

memberService.modifyMemberCondition(strSQL);