风险定价项目系统管理模块详细设计

|  |  |
| --- | --- |
| 系统名称 | 风险定价项目 |
| 模块名称 | 系统管理模块 |
| 项目负责人 | 刘振坤 |
| 模块设计 | 刘振坤，白校正 |
| 文档提交日期 | 2015/12/3 |

共鸣时代科技有限公司

(版权所有,翻版必究)

修改记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | 修改后版本号 | 修改内容简介 | 修改日期 | 修改人 |
| 1 | V1.0 | 创建 | 2015-12-03 | 刘振坤 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# 背景

详见MRD

# 名词解释

# 总体设计

## 实现的子系统

|  |  |  |
| --- | --- | --- |
| # | 子系统名称 | 功能描述 |
| 1 | 风险定价内/外部系统 | 内/外：用户登录系统，添加新型规则，并配置形成新型工作流; 用户可以查看新型工作流/规则列表, 启用或者禁用； 外：实现接口，如审核新型工作流接口; 获取新型工作流/规则列表接口; 获取 |
| 2 | 决策引擎，decision-rpc | 调用外部数据源，遍历新型工作流中的新型规则，运行规则逻辑，产生结果并返回调用方 |

## 系统关系图

## 部署架构

**说明：**

* MFS指分布式文件系统
* Solr需要支持主从复制，一主2从
* Db需要支持主从复制，一主2从
* Redis需要支持复制，一主1从

## 安全设计

1. 使用硬件token登录，pin码+token码组合验证，初次登录时pin码必须被修改
2. CMS系统和优良不良记录系统使用HTTPS，保证传输层安全
3. 索引服务、redis服务部署在内网，不对公网暴露服务
4. CMS系统和优良不良记录系统提供的内部api，使用https+MD5签名+ip白名单保证安全
5. Api的提供方和消费方对相同的数据项进行序列化
6. 双方配置相同的salt，对 序列化串+salt进行md5签名
7. 代码级别安全：

* 使用参数化访问数据库，避免sql注入
* 网站页面显示文章时使用freemarker的html展示函数避免xss

1. 测试保证

* 系统上线前对全部系统进行xss攻击测试

## 系统环境

## 假设及与其它模块联系

暂无

## 相关软件及硬件

1. 软件：

* Linux centos 4.3
* JDK1.7
* Tomcat 6
* Mysql 5.1
* Mongo

1. 硬件：

* 标准配置物理机

## 模块限制

无

## 数据规模估计

无

# 数据库设计

## 外部数据源规则信息表 (outer\_common\_rule)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 备注 | | 是否为空 |
| id | bigInt | 20 | 主键id自增 | | NO |
| name | Varchar | 256 | 规则名称 | | NO |
| source | tinyint | 2 | 来源 | 0:汇法 | NO |
| 1:聚信立 |
| 2:鹏远 |
| 3:前海 |
| 4:算话 |
| 5:同盾 |
| 6:中胜 |
| type | tinyint | 2 | 类型 | 0:黑名单 | NO |
| 1:反欺诈 |
| 2:违法 |
| status | tinyint | 2 | 有效性 | 0:无效 | NO |
| 1:有效 |
| user\_id | bigint | 20 | 创建人ID | | NO |
| org\_id | bigint | 20 | 创建机构ID | | NO |
| create\_time | datetime | - | 创建时间 | | NO |
| update\_time | datetime | - | 更新时间 | | NO |

## 

## 外部数据源工作流信息表 (outer\_common\_workflow)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 备注 | | 是否为空 |
| id | bigInt | 20 | 主键id自增 | | NO |
| name | Varchar | 256 | 工作流名称 | | NO |
| status | tinyint | 2 | 有效性 | 0:无效 | NO |
| 1:有效 |
| user\_id | bigint | 20 | 创建人ID | | NO |
| org\_id | bigint | 20 | 创建机构ID | | NO |
| create\_time | datetime | - | 创建时间 | | NO |
| update\_time | datetime | - | 更新时间 | | NO |

## 外部数据源工作流与规则关系表(outer\_workflow\_rule\_relation)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 备注 | | 是否为空 |
| id | bigInt | 20 | 主键id自增 | | NO |
| status | tinyint | 2 | 有效性 | 0:无效 | NO |
| 1:有效 |
| rule\_id | bigint | 20 | 规则ID | | NO |
| workflow\_id | bigint | 20 | 工作流ID | | NO |
| user\_id | bigint | 20 | 创建人ID | | NO |
| org\_id | bigint | 20 | 创建机构ID | | NO |
| create\_time | datetime | - | 创建时间 | | NO |
| update\_time | datetime | - | 更新时间 | | NO |

# 设计思路及折衷

# 模块设计

## 用户管理

### 功能描述

用户管理，管理使用系统的用户，功能包括用户的添加、更新、启用和禁用功能。可以给用户分配小组和角色以达到权限控制的目的。

### 流程图及说明

### 数据结构及说明

### 算法描述（可选）

* 在权限中心设置权限保存时要从数据库中获取原有权限，默认为空 由超级管理员进行权限分配 每人一次一次分配一个权限 如果没有相同权限 则要把当前分配的权限或上已拥有的权限进行更新。如果以拥有相同的权限 异或上分配的权限进行更新
* 查询是否具有某个权限时先从数据库中拿到，然后使用db的数据跟查询的这个权限与 返回true 或 false
* 获取该用户整个权限时，取出数据库的权限 在方法中 通过逐个与 来判断是否有那个权限返回给程序
* 权限在后台使用 ENUM进行定义 [**public** **enum** ERole](#_类)
* 顶级部门的算法:该人员的path第二位 就是顶级部门的id 如： /0/2/

### 与其他子模块的接口

**权限相关接口**

获取具有审核权限的人员列表

|  |  |  |  |
| --- | --- | --- | --- |
| url | /api/getExamineRole | | |
| 入参 | 参数名 | 类型 | 备注 |
| memberId | Int | 人员id |
| 返回 | [  {“id’:”1”,”name”:”\*\*”},  {“id’:”2”,”name”:”\*\*”},  {“id’:”3”,”name”:”\*\*”}  ] | | 该人员所在部门的所有审核员（具体返回参数待定） |

获取栏目列表及是否拥有某权限

|  |  |  |  |
| --- | --- | --- | --- |
| url | /api/getMemberRole | | |
| 入参 | 参数名 | 类型 | 备注 |
| memberId | Int | 人员id |
| columnCmsName | String | 角色名称 |
| columnCms | Boolean | 是否获取栏目列表 |
| 返回 | [  {  " bear ": "true"  },  {  "column": [  {  "id": "1",  "name": "新闻"  },  {  "id": "2",  "name": "政策"  }  ]  }  ] | | 是否具有该权限，如果获取栏目列表 |

### 与前端的接口

权限操作

|  |  |  |  |
| --- | --- | --- | --- |
| url | /temp/addRoles | | |
| 入参 | 参数名 | 类型 | 备注 |
| roleId | Int | 角色id |
| addMemberId | Int | 新增权限的人员id |
| delMemberId | Int | 删除权限的人员id |
| 返回 | {“bear”:”true/false”} | | 操作成功与否 |

这里有两步操作:

把新增权限的人的权限或上当前权限

把删除权限的人的权限异或上当前权限

## 上传文件

### 功能描述

### 流程图及说明



### 数据结构及说明

### 算法描述（可选）

文件统一存在分布式文件系统下。在存储的时候按年月分不同的文件夹（比如2015年一月份上传的文件存储在“/201501”文件夹下），文件上传完成后进行数据库操作，操作文章附件表（t\_article\_save）,存入id（图片id），name（上面生成的UUID），path（上面生成的文件夹如201501），type（存文件扩展名）

### 与其他子模块的接口

### 与前端的接口

上传接口：文件存储按月份来进行存贮。文件以UUID的形式命名。返回文件图片UUID

|  |  |  |
| --- | --- | --- |
| URL | /cms/upload | |
| 传入参数 | 参数名 | 参数类型 |
| files | MultipartFile |
| 返回值 | 图片UUID | |

## 图片展示

### 功能描述

### 流程及说明

文件显示访问路径统一为”domain/uploads/uuid.扩展名。

CMS中的显示:

假设访问路径为”cms.sdfefadf.cdsfa/uploads/0aebb264-6cb2-4dca-9650-3335efec010d.jpg”



Website前段请求：

假设访问路径为:”www.dgdfs.com/uploads/0aebb264-6cb2-4dca-9650-3335efec010d.jpg”



### 算法描述（可选）

**public** **enum** ERole {

***ADMIN*** (1,"ADMIN","超级管理员"),

***CMSCOLUMN*** (2,"CMSCOLUMN","CMS栏目"),

***MOULDMANAGE*** (4,"MOULDMANAGE","模板"),

***MOULDUPLOAD***(8,"MOULDUPLOAD","模板上传"),

***MOULDEXAMINE***(16,"MOULDEXAMINE","模板审批");

**private** **int** id;

**private** String label;

**private** String zhLabel;

**private** ERole(**int** id, String label, String zhLable) {

**this**.id = id;

**this**.label = label;

**this**.zhLabel = zhLable;

}

**public** **static** ERole getERoleId(**int** id){

**if**(id==1){

**return** ERole.***ADMIN***;

}

**if**(id==2){

**return** ERole.***CMSCOLUMN***;

}

**if**(id==4){

**return** ERole.***MOULDMANAGE***;

}

**if**(id==8){

**return** ERole.***MOULDUPLOAD***;

}

**if**(id==16){

**return** ERole.***MOULDEXAMINE***;

}

**return** **null**;

}

}

### 与其他子模块的接口

### 与前端的接口

上传接口：文件存储按月份来进行存贮。文件以UUID的形式命名。返回文件图片UUID。

|  |  |  |
| --- | --- | --- |
| URL | /cms/upload | |
| 传入参数 | 参数名 | 参数类型 |
| files | MultipartFile |
| 返回值 | 图片UUID | |

## 栏目管理

### 功能描述

本功能具体使用的表：channel

### 流程及说明

### 算法描述（可选）

### 与其他子模块的接口

### 与前端的接口

新建/修改栏目

栏目的新增/修改功能

|  |  |  |  |
| --- | --- | --- | --- |
| url | chanel/saveOrUpdate | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | id | Long | 主键id |
| name | String | 栏目名称（必传） |
| englishName | String | 栏目英文名 |
| type | enum | 栏目类型：(必传)  main(首页)，creditStatus(信用状态)，poilicRegulation(政策法规)，specialReport(专题报告)，  goodRecord(优良记录)，  badRecord(不良记录) |
| 返回值 | [{“code”:200},{“message”,”success”}] | | 创建成功 |

* + 查询栏目

栏目的查询功能

|  |  |  |  |
| --- | --- | --- | --- |
| url | chanel/conditionList | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | type | enum | 栏目类型：  main(首页)，creditStatus(信用状态)，poilicRegulation(政策法规)，specialReport(专题报告)，  goodRecord(优良记录)，  badRecord(不良记录) |
| 返回值 | [{“code”:200},  {“message”,”success”},  {“data”:[  {  “id” : 1,  “name”:”最新消息”,  “englishName”:”newMessage”,  “type”:” creditStatus”,  “status”: “yes”  },  {  “id” : 2,  “name”:”国内信用”,  “englishName”:”chinaCredit”,  “type”:” creditStatus”,  “status”: “no”  }  ]}] | | 说明：  status: 枚举类型  yes表示启用  no 表示未启用/禁用 |

状态启用/禁用

栏目状态的启用/禁用功能

|  |  |  |  |
| --- | --- | --- | --- |
| url | chanel/updateStatus | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | id | Long | 主键id |
| status | enum | 栏目类型：(必传)  yes 启用  no 未启用 |
| 返回值 | [{“code”:200},{“message”,”success”}] | | 状态修改成功 |

根据ID栏目查询

栏目的查询功能

|  |  |  |  |
| --- | --- | --- | --- |
| url | chanel/get | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | id | Long | 主键id |
| 返回值 | [{“code”:200},  {“message”,”success”},  {“data”:  {  “id” : 1,  “name”:”最新消息”,  “englishName”:”newMessage”,  “type”:” creditStatus”,  “status”: “yes”  }  }] | | 说明：  status: 枚举类型  yes表示启用  no 表示未启用/禁用 |

## 文章管理

### 功能描述

本功能具体使用的表：article，article\_attach，article\_content

### 流程及说明

新建文章执行下图流程



### 算法描述（可选）

### 与其他子模块的接口

### 与前端的接口

上传功能参照文件上传模块

文章的新增/修改功能

|  |  |  |  |
| --- | --- | --- | --- |
| url | artice/saveOrUpdate | | |
| 类型定义 | showType定义是枚举：   * title(头条图)， * focalPonint(焦点图)， * inernal(内部图)  |  |  | | --- | --- | |  |  | | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | id | Long | 主键id |
| titleP | String | 主标题 |
| titleF | String | 副标题 |
| englishTitle | String | 英文标题 |
| abstract | String | 文章摘要 |
| content | String | 文章内容 |
| chanel\_id | Long | 所属栏目类型 |
| showTypeList | List | Uuid和showtype的集合 |
| uuidList | List | 存放返回图片UUID的集合 |
| isFocalPoint | Boolean | 是否是重点推荐文章 |
| 返回值 | [{“code”:200},{“message”,”success”}] | | 创建/修改成功 |

文章查询

文章的查询

|  |  |  |  |
| --- | --- | --- | --- |
| url | artice/conditionList | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | type | enum | 栏目类型：  main(首页)，creditStatus(信用状态)，poilicRegulation(政策法规)，specialReport(专题报告)，  goodRecord(优良记录)，  badRecord(不良记录) |
| 返回值 | [{“code”:200},{“message”,”success”},  {“data”:  {  “id”:1,  “titleP”:”起火”,  “titleF”:”XX大厦起火”,  “englishTitle”:””,  “abstract”:”XXX地方起火”,  “name”:”最新消息”,  “type”:”政策法规”,  “isFocalPoint”:false  },  {  “id”:2,  “titleP”:”工作条例”,  “titleF”:” 人民检察院监察工作条例”,  “englishTitle”:””,  “abstract”:”XXX地方起火1”,  “name”:”政策法规”,  “type”:”政策法规”,  “isFocalPoint”:false  }  ] | |  |

文章发布

文章的发布功能

|  |  |  |  |
| --- | --- | --- | --- |
| url | artice/release | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | idList | List | 主键id集合 |
| 返回值 | [{“code”:200},{“message”,”success”}] | | 发布成功 |

文章删除

文章删除功能

|  |  |  |  |
| --- | --- | --- | --- |
| url | artice/delete | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | id | Long | 主键id |
| 返回值 | [{“code”:200},{“message”,”success”}] | | 删除成功 |

文章恢复

文章恢复功能

|  |  |  |  |
| --- | --- | --- | --- |
| url | artice/recovery | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | id | Long | 主键id |
| 返回值 | [{“code”:200},{“message”,”success”}] | | 恢复成功 |

文章ID查看

文章根据id

|  |  |  |  |
| --- | --- | --- | --- |
| url | artice/get | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | id | Long | 主键id |
| 返回值 | [{“code”:200},{“message”,”success”},  {“data”:  {  “id”:1,  “titleP”:”起火”,  “titleF”:”XX大厦起火”,  “englishTitle”:””,  “abstract”:”XXX地方起火”,  “content”: ”XXXXXXXX…….”,  “type”:”政策法规”,  [  {  “id”:1,  “showType”:” inernal”,  “path”:”uuid.jpg”,  },  {  “id”:2,  “showType”:” inernal”,  “path”:”uuid.jpg”,  },  ]  “isFocalPoint”:false  }  ] | | 说明：  showType类型：  title(头条图)，  focalPonint(焦点图)，  inernal(内部图)  isFocalPoint类型：  ture表示是重点，  false表示不是重点 |

## 头条管理

### 功能描述

### 流程及说明

**新建头条执行下图流程**



### 算法描述（可选）

### 与其他子模块的接口

### 与前端的接口

头条的新增/修改功能

|  |  |  |  |
| --- | --- | --- | --- |
| url | title/saveOrUpdate | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | id | Long | 主键id |
| articleId | Long | 文章id |
| name | String | 存放返回图片UUID |
| showType | enum | show类型描述  title(头条图)，  focalPonint(焦点图)，  inernal(内部图) |
| descript | String | 描述 |
| 返回值 | [{“code”:200},{“message”,”success”}] | | 创建/修改成功 |

Title查询

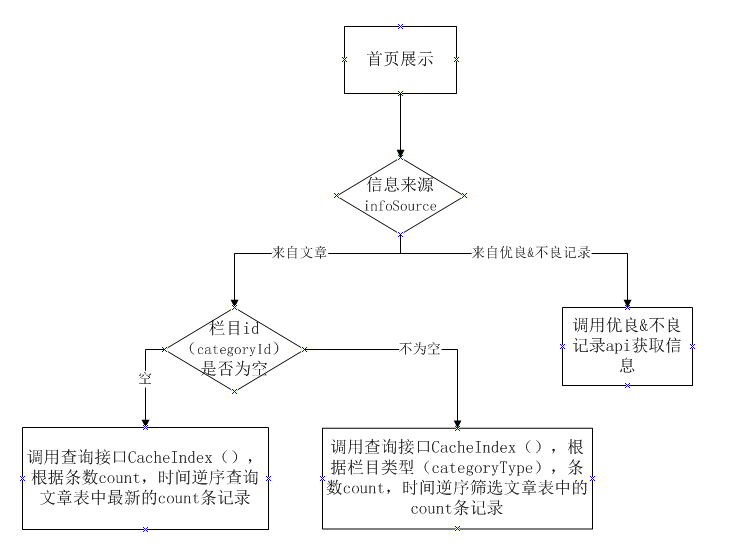
|  |  |  |  |
| --- | --- | --- | --- |
| url | title/list | | |
| 传参 | 参数名 | 类型 | 备注 |
|  | null |  |  |
| 返回值 | [{“code”:200},{“message”,”success”},  {“data”:  {  “id”:1,  “articleTitleP”:”外滩事件”,  “path”:”url”  },  {  “id”:2,  “articleTitleP”:”讨薪死亡事件”,  “path”:”url”  }  ] | | Path描述：  表示图片路径 |

## Block信息展示

### Freemarker指令参数

|  |  |
| --- | --- |
| Name | 含义 |
| blockName | block的name。保证每个block的name网站内唯一 |
| infoSource | 信息来源（文章或者优良&不良记录） |
| categoryId | 栏目id |
| categoryType | 栏目类型（焦点，小图，重点，头条） |
| count | 本次查询信息的条数 |

### 流程及说明



### 实现描述（可选）

interface BlockDataProvider{

String getData(blockName, categoryId, categoryType, count);

}

参数：

|  |  |
| --- | --- |
| Name | 含义 |
| blockName | block的name。保证每个block的name网站内唯一 |
| categoryId | 栏目id |
| categoryType | 栏目类型（焦点，小图，重点，头条） |
| count | 本次查询信息的条数 |

接口实现：

两个实现类

// 第一个实现类

class DB BlockDataProvider Impl implements BlockDataProvider {

    // 实现了接口中定义的方法，必须用关键字public因为接口中定义的方法默认是public的。

public String indexInfo(blockName, categoryId, categoryType, count) {

从db中查询信息

}

}

// 第二个实现类

class CachedBlockDataProvider Impl implements BlockDataProvider {

    // 实现了接口中定义的方法，必须用关键字public因为接口中定义的方法默认是public的。

public String indexInfo(blockName, categoryId, categoryType, count) {

根据blockName先从cache（redis）中查询是否有对应的结果String，

有则直接返回结果，没有则去db中查询信息，并且把k值blockName，结果value 存入cache中

            }

}

### 与其他子模块的接口

### 与前端的接口

## 优良不良记录

### 功能描述

### 流程及说明

新建（导入）优良&不良记录



### 实现描述（可选）

优良&不良记录索引创建

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Index(是否索引) | Store（是否存储） | 分词（是否需要分词） |  |
| id(记录主键,solr主键) | N | Y | N |  |
| tid（模板id） | Y | Y | N |  |
| company(公司名称) | Y | Y | Y(IK分词) |  |
| type（优良&不良） | Y | Y | N |  |
| content(values) | Y | N | Y(IK分词) |  |

优良&不良记录索引更新

使用记录主键（id）字段作为solr schame主键

当修改记录后，重新进入审核流程，审批完成后，该记录对应的索引文件将同步更新

当删除记录后，异步删除对应索引信息

以优良记录检索示例（不良记录检索类似）

前端传后端检索关键词 、 记录类型标记 、pagesize、page（具体接口参见前后端接口）

后端检索逻辑：（关键词匹配公司名称 or 关键词匹配记录内容）and type=记录类型

返回前端记录list & 总记录数 列表显示公司名称(去重)

当用户点击具体公司链接，根据名称反查数据库返回前端该公司优良记录，前端页面渲染

### 与其他子模块的接口

### 与前端的接口

新建模板接口：

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | /template/add | | |
| 请求时机 | 新建模板接口 | | |
| 功能 | 新建模板接口 | | |
| 请求参数 | name | 模板名称 | |
| departmentId | 部门id | |
| templateType | 模板类型1:不良模板 2:优良模板 3:基础信息 | |
| creditType | 信用类型 template\_type=1时 1:行政裁决 2:行政处罚;template\_type=2时 3:行政奖励 | |
| shardingLevel | 共享程度 1:社会共享 | |
| templateManager | 模板创建者（存id or 存名称） | |
| items | List<String>  a#aa#a#a#a#a#a#a | |
| auditer | 模板审批者，多个|分割 传审批者id | |
| 响应 | {  "data": {},  "msg": [],  "status": 0,  "statusInfo": {}  } | |  |

批量修改模板状态：

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | template/updateStatus | | |
| 请求时机 | 批量启动模板 | | |
| 功能 | 批量启动模板 | | |
| 请求参数 | tids | List<Integer> | |
| status | 1:启用 2:删除 | |
| 响应 | {  "data": {},  "msg": [],  "status": 0,  "statusInfo": {}  } | |  |

模板查询：

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | template/query | | |
| 请求时机 | 过滤搜索模板 | | |
| 功能 |  | | |
| 请求参数 | departmentId | 部门id（所有部门时可传-1）int | |
| creditType | 信用类型（所有类型时传-1）int | |
| query | 关键词搜索（可选） string | |
|  | page | 当前查询页码 | |
|  | pageSize | 每页大小 | |
| 响应 | {  "data": {list:[{tid:11,name:aa,department:”万恶的共产党”,creditType:”行政处罚”,status:1,templateManager:”张三”,isUser：1，addTime:” 2014\_12\_31 12:20:10”},{}]  },  "msg": [],  "status": 0,  "statusInfo": {}  } | | isUser 0:模板还未使用  1:模板已使用  如果模板已被使用，前端可控制为不允许修改，但可删除（后端只逻辑删除，以便已使用该模板的记录可获取模板信息） |
|  |  | |  |

获取模板对应的items：

|  |  |  |
| --- | --- | --- |
| 请求URL | /template/queryItemsByTid | |
| 请求时机 | 查询单个模板对应的items | |
| 功能 |  | |
| 请求参数 | tid | int |
| 响应 | {  "data": { list:[  {name:案号,type:单行文本,length:200,required:0,displayType:0,status:0,isCompanyName:0  }，  {}，  {}  ]  },  "msg": [],  "status": 0,  "statusInfo": {}  } |  |
|  |  |  |

模板修改接口：

|  |  |  |
| --- | --- | --- |
| 请求URL | /template/update | |
| 请求时机 | 修改模板 | |
| 功能 | 只有该模板还未被使用的情况下才允许修改（mysql无法做到绝对的，但保存时可再次判断一下） | |
| 请求参数 | 1. **请求参数与add是一致** | 注意此处是模板信息全部更新 |
| 响应 | {  "data": {  },  "msg": [],  "status": 0,  "statusInfo": {}  } |  |

## 优良&不良记录管理

### 功能描述

### 流程及说明

### 算法描述（可选）

### 与其他子模块的接口

### 与前端的接口

根据模板查询对应记录接口：

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | /record/getAllRecordsByTid | | |
| 请求时机 | 根据模板id获取 | | |
| 功能 | 根据模板id获取 | | |
| 请求参数参数 | tid | 模板id | |
| startTime | 查询起始时间（可选），没有为空 | |
| endTime | 查询结束时间（可选），没有为空 | |
| status | 0:已发布；1:待审批；2：被拒绝；3：暂存（新建） | |
| query | 查询公司名称（模糊匹配） | |
| page | 页码 | |
| pageSize | 每页大小 | |
| 响应 | {  "data": {  "list": [  {    }  ]，  templateName:”被执行人名单”，  departmentName:”操蛋委”  },  "msg": [],  "status": 0,  "statusInfo": {}  } | | 根据不同的模板返回不同的map 对应的json，前端直接显示 key : value |

新增记录

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | record/add | | |
| 请求时机 | 新增优良&不良记录 | | |
| 功能 | 新增优良&不良记录 | | |
| 请求参数 | tid | 模板id int | |
| company | 公司名称（必填），前端在得到的模板格式中，有个字段标记模板项哪个是公司名称（只有一个），当发现这个字段后，立即通过company属性给后端 | |
| messageType | 冗余存储（方便查询） 1:不良记录,2:优良记录,3:其他 | |
| content | String 前端返回给后端拼好的json字符串 | |
| status | 标记是暂存or提交（3:暂存 1:待审批） | |
| 响应 | {  "data": {    },  "msg": [],  "status": 0,  "statusInfo": {}  } | |  |

修改记录

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | record/update | | |
| 请求时机 | 修改优良&不良记录 | | |
| 功能 | 修改优良&不良记录 | | |
| 请求参数 | mid | 记录id | |
| company | 公司名称（必填），前端在得到的模板格式中，有个字段标记模板项哪个是公司名称（只有一个），当发现这个字段后，立即通过company属性给后端 | |
| content | String 前端返回给后端拼好的json字符串 | |
| status | 标记是暂存or提交（3:暂存 1:待审批） | |
| 响应 | {  "data": {    },  "msg": [],  "status": 0,  "statusInfo": {}  } | |  |

批量提交记录进入审核 or 批量删除记录（根据status字段来区分）

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | record/update | | |
| 请求时机 | 只有新建（暂存）状态的记录才能批量提交进入审核（前端需要做控制），后端也许做判断 | | |
| 功能 | 对新建（暂存）状态记录进行提交 | | |
| 请求参数 | mids | 记录id集合 List<Integer> | |
| status | 1:批量提交流程（进入待审核状态）  4:批量删除 | |
| 响应 | {  "data": {    },  "msg": [],  "status": 0,  "statusInfo": {}  } | |  |

批量审核记录（通过或不通过）

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | record/audit | | |
| 请求时机 | 审核员登录系统后，查询其下待审核的记录列表 | | |
| 功能 | 对新建（暂存）状态记录进行提交 | | |
| 请求参数 | mids | 记录id集合 List<Integer> | |
| status | 0:通过  2:不通过 | |
|  | comment | 拒绝理由必填，通过可选 | |
| 响应 | {  "data": {    },  "msg": [],  "status": 0,  "statusInfo": {}  } | |  |

## 网站展现使用api

### 功能描述

### 流程及说明

### 算法描述（可选）

### 与其他子模块的接口

### 与前端的接口

* 查询某企业下所有的优良or不良记录

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | record/getRecordsByName | | |
| 请求时机 | 全站搜索出结果后，进一步查看每个结果的详细信息 | | |
| 功能 | 如果不做id映射，此处可直接将结果返回前端展现，不需要再次查询其模板信息 | | |
| 请求参数 | mids | 记录id集合 List<Integer> | |
| status | 0:通过  2:不通过 | |
|  | comment | 拒绝理由必填，通过可选 | |
| 响应 | {  "data": {  list:[{id:1,tid:12,company:”百度在线”,content:{法人:”robin”，奖励金额：1}}]  },  "msg": [],  "status": 0,  "statusInfo": {}  } | |  |

* 获取最近发布的优良或不良信息

给网站返回最近更新的优良or不良记录（以更新时间为准）条数暂停3条

|  |  |  |  |
| --- | --- | --- | --- |
| 请求URL | record/ | | |
| 请求时机 | 网站首页需获取n条最新审核通过的记录 | | |
| 权限校验 | 具体验证方案待细化 | | |
| 功能 | 网站首页需获取n条最新审核通过的记录 | | |
| 请求参数 | type | 可选填。1：不良记录 2 优良记录 | |
| 响应 | {  "data": {  list:[{id:1,tid:12,company:”百度在线”,content:{法人:”robin”，奖励金额：1}}]    },  "msg": [],  "status": 0,  "statusInfo": {}  } | |  |

* 根据记录id获取记录详情

|  |  |  |
| --- | --- | --- |
| 请求URL | record/getDetailById | |
| 请求时机 | 根据记录id获取记录的详细信息 | |
| 权限校验 | 具体验证方案待细化 | |
| 功能 | 后端结合记录模板信息返回前端直接可展现的json | |
| 请求参数 | id | 记录id |
|  | 权限验证信息 |  |
| 响应 | {  "data": {  list:[{id:1,tid:12,company:”百度在线”,content:{法人:”robin”，奖励金额：1}}]    },  "msg": [],  "status": 0,  "statusInfo": {}  } | |

## 日志管理

### 功能描述

### 流程及说明

### 实现描述（可选）

* 创建单独的日志库
* 异步记录，使用activeMQ记录日志
* 创建aop拦截器和注解，自动记录日志

### 与其他子模块的接口

### 与前端的接口

# 风险评估及对其它模块/系统影响（可选）

## 已知的或可预知的风险

在这里加上已经知道的或可能会发生的风险，包括技术、业务等方面。最好针对每个风险，列出相应的应对措施。

## 与其它模块/系统可能的影响

在“4.1 ”中描述了该模块与其它模块的依赖关系。在这里描述这些依赖关系可能带来的影响。包括本模块对其它模块可能造成的影响以及其它模块可能给本模块造成的影响两个方面。

# 设计评审意见

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | jjkkjj问题描述 | 提出人 | 处理方式/说明 | 状态 |
| 1 |  |  |  | Open |
| 2 |  |  |  | Close |
| 3 |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

这里累计记录和跟踪该文档评审时发现的问题。

# 附件及参考资料

填写文档相关的附件或参考资料。若是不常变更的文档（比如调研报告），建议以对象方式插入到该文档中。如是经常变更的文档（比如接口文档），建议在此列出文件名即可。