**HDI一期详细设计说明书V1.0**

**广东以大供应链管理有限公司**

**编写人：罗荣熙**

**2019.04.22**

修改记录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 版本号 | 内容 | 修改人 | 审批人 | 发布日期 |
| 1 | V1.0 | 创建：添加供应商信息设计 | 罗荣熙 | 罗然熹 | 2019.04.22 |
| 2 | V1.1 | 修改：添加厂商信息设计 | 罗荣熙 | 罗然熹 | 2019.04.24 |
| 3 | V1.2 | 修改：添加字典查询设计 | 罗荣熙 | 罗然熹 | 2019.04.26 |
| 4 | V1.3 | 修改：添加系统消息设计 | 罗荣熙 | 罗然熹 | 2019.04.28 |
| 5 | V1.4 | 修改：添加代理商信息设计 | 罗荣熙 | 罗然熹 | 2019.04.29 |
| 6 | V1.5 | 修改：添加平台药品、供应商药品、医院药品设计 | 罗荣熙 | 罗然熹 | 2019.05.09 |

目录

[1. 引言 1](#_Toc23179)

[1.1编写的目的 1](#_Toc31803)

[1.2术语表 1](#_Toc7570)

[1.3参考资料 1](#_Toc30025)

[2. 工具与规范 1](#_Toc3619)

[2.1开发工具 1](#_Toc10759)

[2.2代码规范 1](#_Toc4016)

[3. 平台结构图 1](#_Toc11430)

[4. 数据结构说明 1](#_Toc15911)

[4.1机构管理 1](#_Toc6633)

[4.1.1表清单 1](#_Toc14724)

[4.1.2表结构 1](#_Toc22274)

[4.2统一编码管理 1](#_Toc2764)

[4.2.1表清单 1](#_Toc1809)

[4.2.2表结构 1](#_Toc22024)

[4.3商品管理 1](#_Toc15806)

[4.3.1表清单 1](#_Toc26878)

[4.3.2表结构 1](#_Toc24457)

[4.4证照管理 1](#_Toc29873)

[4.4.1表清单 1](#_Toc26059)

[4.4.2表结构 1](#_Toc28663)

[4.5采购管理 1](#_Toc11126)

[4.5.1表清单 1](#_Toc21410)

[4.5.2表结构 1](#_Toc20755)

[4.6供货管理 1](#_Toc9489)

[4.6.1表清单 1](#_Toc31131)

[4.6.2表结构 1](#_Toc10785)

[4.7退货管理 1](#_Toc28436)

[4.7.1表清单 1](#_Toc14808)

[4.7.2表结构 1](#_Toc29982)

[4.8系统消息 1](#_Toc30535)

[4.8.1表清单 1](#_Toc206)

[4.8.2表结构 1](#_Toc28772)

[5. 模块设计 1](#_Toc2191)

[5.1机构管理 1](#_Toc10491)

[5.1.1供应商信息 1](#_Toc29664)

[5.1.2医院信息 1](#_Toc6630)

[5.1.3厂商信息（平台） 1](#_Toc12838)

[5.1.4代理商信息 1](#_Toc12065)

[5.2统一编码管理 1](#_Toc30958)

[5.2.1用例图 1](#_Toc25472)

[5.2.1功能设计说明 **错误!未定义书签。**](#_Toc27980)

[5.3商品管理 1](#_Toc20238)

[5.3.1平台药品信息 1](#_Toc22853)

[5.3.2供应商药品信息 1](#_Toc25946)

[5.3.3医院药品信息 1](#_Toc21853)

[5.3.4平台试剂信息 1](#_Toc19263)

[5.3.5供应商试剂信息 1](#_Toc20418)

[5.3.6医院试剂信息 1](#_Toc10226)

[5.3.7平台耗材信息 1](#_Toc26029)

[5.3.8供应商耗材信息 1](#_Toc13131)

[5.3.9医院耗材信息 1](#_Toc26864)

[5.4证照管理 1](#_Toc5043)

[5.4.1用例图 1](#_Toc31433)

[5.4.2功能设计说明 1](#_Toc28695)

[5.5采购管理 1](#_Toc11888)

[5.5.1用例图 1](#_Toc32458)

[5.5.2功能设计说明 1](#_Toc30975)

[5.6供货管理 1](#_Toc24649)

[5.6.1用例图 1](#_Toc8606)

[5.6.2功能设计说明 1](#_Toc19765)

[5.7退货管理 1](#_Toc15392)

[5.7.1用例图 1](#_Toc6399)

[5.7.1功能设计说明 1](#_Toc18801)

[5.8公共模块 1](#_Toc22300)

[5.8.1字典查询设计 1](#_Toc13444)

[5.8.2枚举查询设计 1](#_Toc7872)

[5.8.3导入功能设计 1](#_Toc22435)

[5.8.4导出功能设计 1](#_Toc2405)

[5.9系统消息 1](#_Toc21839)

[5.9.1系统消息管理 1](#_Toc10931)

[5.9.2用户消息管理 1](#_Toc10514)

# 引言

## 1.1编写的目的

本详细设计说明书编写的目的是说明程序模块的设计考虑，包括程序描述、输入/输出、算法和流程逻辑等，为软件编程和系统维护提供基础。本数米数的预期读者为系统设计人员、软件开发人员、软件测试人员和项目评审人员。

## 1.2术语表

|  |  |  |
| --- | --- | --- |
| 序号 | 术语或缩略语 | 说明行定义 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## 1.3参考资料

|  |  |  |  |
| --- | --- | --- | --- |
| 资料名称 | 作者 | 文件编号、版本 | 资料存放地址 |
| HDI总体设计方案 | 罗荣熙 | V1.0 | svn://192.168.1.12/develop/hscm/hdi/trunk/hdi/hdi-document/技术方案 |
| HDI一期技术设计方案 | 罗荣熙 | V1.1 | svn://192.168.1.12/develop/hscm/hdi/trunk/hdi/hdi-document/技术方案 |
| HDI一期解决方案 | 郑丽君 | V1.3 | svn://192.168.1.12/develop/hscm/hdi/trunk/hdi/hdi-document/需求分析 |
| HDI一期研发工作计划 | 罗荣熙 | V1.0 | svn://192.168.1.12/develop/hscm/hdi/trunk/hdi/hdi-document/工作计划 |
| HDI库表设计 | 罗荣熙 |  | svn://192.168.1.12/develop/hscm/hdi/trunk/hdi/hdi-document/库表设计 |

# 工具与规范

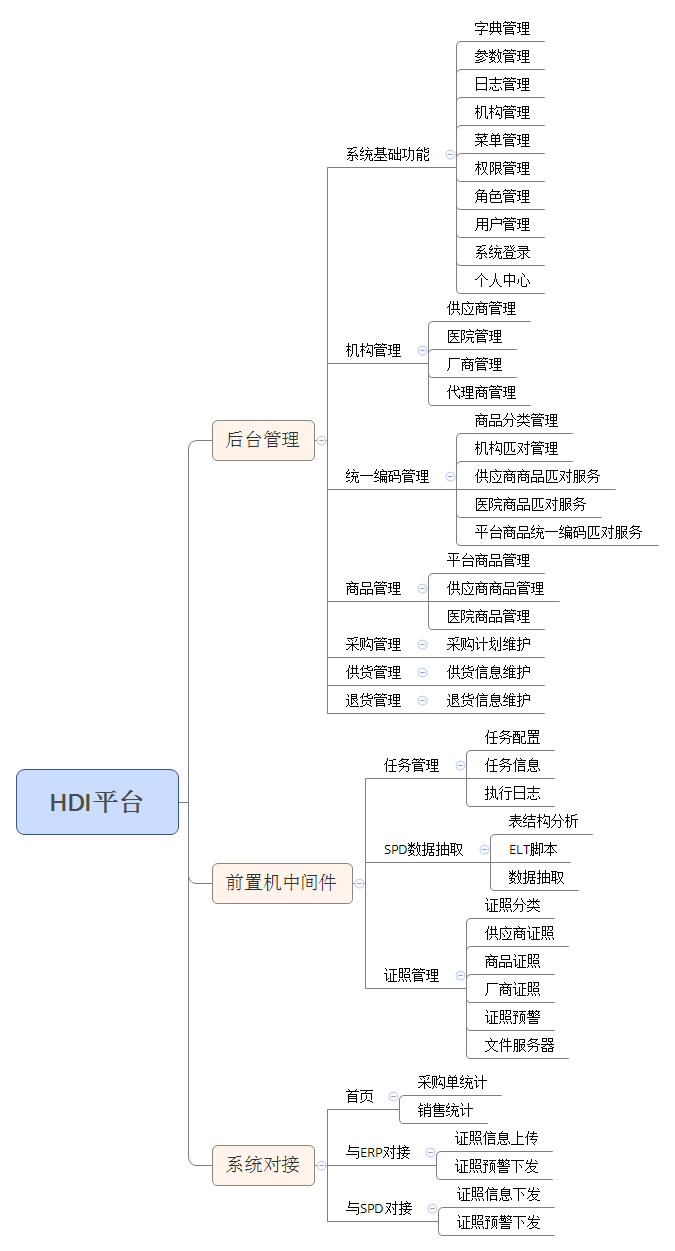
## 2.1开发工具

|  |  |  |
| --- | --- | --- |
| 工具 | 版本 | 集成插件 |
| jdk | 1.8以上 |  |
| eclipse | 4.6.0以上 | Lombok |
| spring tool suite |
| mysql | 5.5.38以上 |  |
| maven | 3.3.3以上 |  |
| powerdesigner | 16以上 |  |

## 2.2代码规范

|  |  |  |  |
| --- | --- | --- | --- |
| 规范 | 内容 | 样例 | 说明 |
| 库表设计 | 表名 | hdi\_org\_supplier\_info | hdi为产品名；org为功能模块名；supplier\_info为具表存储内容；中间用下滑杠隔开，目的是提高可读性。 |
| 字段 | supplier\_code | 中间用下滑杠隔开，目的是提高可读性。 |
| 编码规范 | 代码编写 | supplierCode | 采用驼峰型，目的是提高可读性。 |
| 插入 | insert | 单条记录保存操作 |
| 批量插入 | insertBatch | 批量保存操作 |
| 删除 | deleteBy... | 根据条件进行删除 |
| 更新 | updateBy... | 根据条件进行更新 |
| 条件查询 | selectBy... | 更加条件进行查询 |
| 分页查询 | queryPage | 分页条件进行查询 |
| 状态码 | 系统异常 | -1 |  |
| 请求成功 | 0 |  |
| 请求失败 | 500 |  |

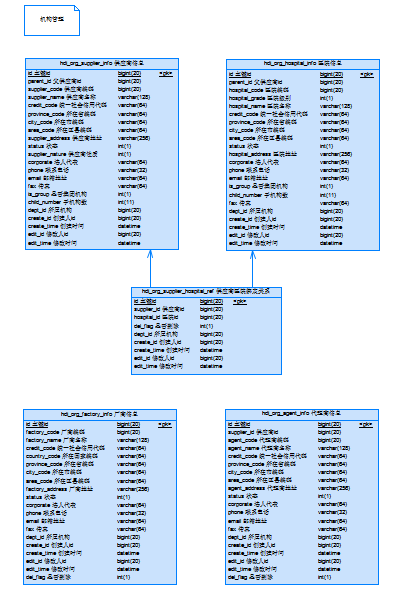
# 平台结构图



# 数据结构说明

包括数据结构名称，功能说明，具体数据结构说明（定义、注释、取值）等。

## 4.1机构管理



### 4.1.1表清单

|  |  |
| --- | --- |
| 名称 | 说明 |
| hdi\_org\_supplier\_info | 供应商信息 |
| hdi\_org\_hospital\_info | 医院信息 |
| hdi\_org\_supplier\_hospital\_ref | 供应商医院绑定关系 |
| hdi\_org\_factory\_info | 厂商信息（平台） |
| hdi\_org\_agent\_info | 代理商信息 |

### 4.1.2表结构

#### 4.1.2.1表hdi\_org\_supplier\_info供应商信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| parent\_id 父供应商id | parent\_id | 父供应商id | bigint(20) |
| supplier\_code 供应商编码 | supplier\_code | 供应商编码(所属区域编码+自编码规则生成) | bigint(20) |
| supplier\_name 供应商名称 | supplier\_name | 供应商名称 | varchar(128) |
| credit\_code 统一社会信用代码 | credit\_code | 统一社会信用代码 | varchar(64) |
| province\_code 所在省编码 | province\_code | 所在省编码 | varchar(64) |
| city\_code 所在市编码 | city\_code | 所在市编码 | varchar(64) |
| area\_code 所在区县编码 | area\_code | 所在区县编码 | varchar(64) |
| supplier\_address 供应商地址 | supplier\_address | 供应商地址 | varchar(256) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| supplier\_nature 供应商性质 | supplier\_nature | 供应商性质(0:国营企业;1:民营企业;2:中外合资企业;3:股份制企业;4:个体企业) | int(1) |
| corporate 法人代表 | corporate | 法人代表 | varchar(64) |
| phone 联系电话 | phone | 联系电话 | varchar(32) |
| email 邮箱地址 | email | 邮箱地址 | varchar(64) |
| fax 传真 | fax | 传真 | varchar(64) |
| is\_group 是否集团机构 | is\_group | 是否集团机构(0:否;1:是) | int(1) |
| child\_number 子机构数 | child\_number | 子机构数(集团供应商允许创建的子供应商数) | int(11) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.1.2.2表hdi\_org\_hospital\_info医院信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| parent\_id 父供应商id | parent\_id | 父供应商id | bigint(20) |
| hospital\_code 医院编码 | hospital\_code | 医院编码(所属区域编码+自编码规则生成) | bigint(20) |
| hospital\_grade 医院级别 | hospital\_grade | 医院级别(0:三级特等;1:三级甲等;2:三级乙等;3:三级丙等;4:二级甲等;5:二级乙等;6:二级丙等;7:一级甲等;8:一级乙等;9:一级丙等) | int(1) |
| hospital\_name 医院名称 | hospital\_name | 医院名称 | varchar(128) |
| credit\_code 统一社会信用代码 | credit\_code | 统一社会信用代码 | varchar(64) |
| province\_code 所在省编码 | province\_code | 所在省编码 | varchar(64) |
| city\_code 所在市编码 | city\_code | 所在市编码 | varchar(64) |
| area\_code 所在区县编码 | area\_code | 所在区县编码 | varchar(64) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| hospital\_address 医院地址 | hospital\_address | 医院地址 | varchar(256) |
| corporate 法人代表 | corporate | 法人代表 | varchar(64) |
| phone 联系电话 | phone | 联系电话 | varchar(32) |
| email 邮箱地址 | email | 邮箱地址 | varchar(64) |
| is\_group 是否集团机构 | is\_group | 是否集团机构(0:否;1:是) | int(1) |
| child\_number 子机构数 | child\_number | 子机构数(集团医院允许创建的子医院数) | int(11) |
| fax 传真 | fax | 传真 | varchar(64) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.1.2.3表hdi\_org\_supplier\_hospital\_ref供应商医院绑定关系

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| supplier\_id 供应商id | supplier\_code | 供应商id | bigint(20) |
| hospital\_id 医院id | hospital\_code | 医院id | bigint(20) |
| del\_flag 是否删除 | del\_flag | 是否删除(-1:已删除;0:正常) | int(1) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

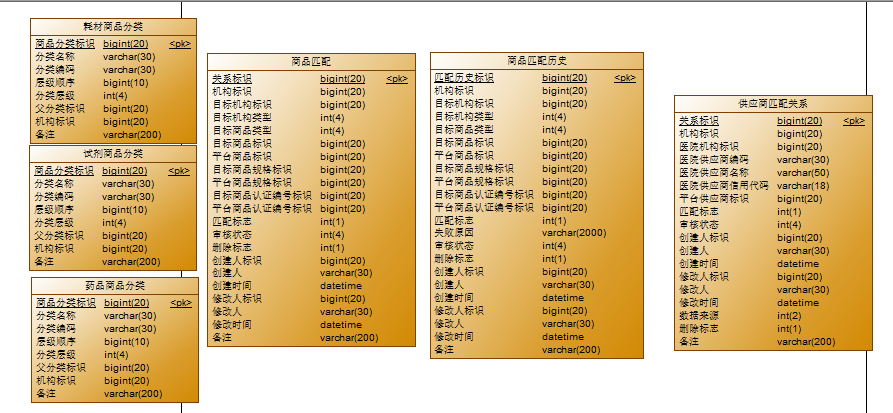
#### 4.1.2.4表hdi\_org\_factory\_info厂商信息（平台）

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| factory\_code 厂商编码 | factory\_code | 厂商编码(所属区域编码+自编码规则生成) | bigint(20) |
| factory\_name 厂商名称 | factory\_name | 厂商名称 | varchar(128) |
| credit\_code 统一社会信用代码 | credit\_code | 统一社会信用代码 | varchar(64) |
| province\_code 所在省编码 | province\_code | 所在省编码 | varchar(64) |
| city\_code 所在市编码 | city\_code | 所在市编码 | varchar(64) |
| area\_code 所在区县编码 | area\_code | 所在区县编码 | varchar(64) |
| factory\_address 厂商地址 | factory\_address | 厂商地址 | varchar(256) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| corporate 法人代表 | corporate | 法人代表 | varchar(64) |
| phone 联系电话 | phone | 联系电话 | varchar(32) |
| email 邮箱地址 | email | 邮箱地址 | varchar(64) |
| fax 传真 | fax | 传真 | varchar(64) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |
| del\_flag 是否删除 | del\_flag | 是否删除(-1:已删除;0:正常) | int(1) |

#### 4.1.2.5表hdi\_org\_agent\_info代理商信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| agent\_code 代理商编码 | agent\_code | 代理商编码(所属区域编码+自编码规则生成) | bigint(20) |
| agent\_name 代理商名称 | agent\_name | 代理商名称 | varchar(128) |
| credit\_code 统一社会信用代码 | credit\_code | 统一社会信用代码 | varchar(64) |
| province\_code 所在省编码 | province\_code | 所在省编码 | varchar(64) |
| city\_code 所在市编码 | city\_code | 所在市编码 | varchar(64) |
| area\_code 所在区县编码 | area\_code | 所在区县编码 | varchar(64) |
| agent\_address 代理商地址 | agent\_address | 代理商地址 | varchar(256) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| corporate 法人代表 | corporate | 法人代表 | varchar(64) |
| phone 联系电话 | phone | 联系电话 | varchar(32) |
| email 邮箱地址 | email | 邮箱地址 | varchar(64) |
| fax 传真 | fax | 传真 | varchar(64) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |
| del\_flag 是否删除 | del\_flag | 是否删除(-1:已删除;0:正常) | int(1) |

## 4.2统一编码管理



### 4.2.1表清单

|  |  |
| --- | --- |
| 名称 | 说明 |
| hdi\_unicode\_supply\_ship | 供应商匹配关系 |
| hdi\_unicode\_goods\_ship | 商品匹配 |
| hdi\_unicode\_consumables\_cate | 耗材商品分类 |
| hdi\_unicode\_drugs\_cate | 药品商品分类 |
| hdi\_unicode\_reagent\_cate | 试剂商品分类 |

### 4.2.2表结构

#### 4.2.2.1表hdi\_unicode\_supply\_ship 供应商匹配关系

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| 关系标识 | ship\_id | 关系标识 | bigint(20) |
| 机构标识 | dept\_id | 机构标识 | bigint(20) |
| 医院机构标识 | horg\_id | 医院机构标识 | bigint(20) |
| 医院供应商编码 | supply\_no | 医院供应商编码 | varchar(30) |
| 医院供应商名称 | supply\_name | 医院供应商名称 | varchar(50) |
| 医院供应商信用代码 | supply\_code | 医院供应商信用代码 | varchar(18) |
| 平台供应商标识 | porg\_id | 平台供应商标识 | bigint(20) |
| 匹配标志 | ship\_flag | 0 未匹配 1已匹配 | int(1) |
| 创建人标识 | cremanid | 创建人标识 | bigint(20) |
| 创建人 | cremanname | 创建人 | varchar(30) |
| 创建时间 | credate | 创建时间 | datetime |
| 修改人标识 | editmanid | 修改人标识 | bigint(20) |
| 修改人 | editmanname | 修改人 | varchar(30) |
| 修改时间 | editdate | 修改时间 | datetime |
| 数据来源 | datasource | 1接口 2手工 | int(2) |
| 删除标志 | del\_flag | 0 否 1是 | int(1) |
| 备注 | memo | 备注 | varchar(200) |

#### 4.2.2.2表hdi\_unicode\_goods\_ship 商品匹配

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| 关系标识 | ship\_id | 关系标识 | bigint(20) |
| 机构标识 | dept\_id | 机构标识 | bigint(20) |
| 目标机构标识 | torg\_id | 目标机构标识 | bigint(20) |
| 目标机构类型 | torg\_type | 目标机构类型 | int(4) |
| 目标商品类型 | tgoods\_type | 1 药品 2 试剂 3 耗材 | int(4) |
| 目标商品标识 | tgoods\_id | 目标商品标识 | bigint(20) |
| 平台商品标识 | pgoods\_id | 平台商品标识 | bigint(20) |
| 目标商品规格标识 | tspecs\_id | 目标商品规格标识 | bigint(20) |
| 平台商品规格标识 | pspecs\_id | 平台商品规格标识 | bigint(20) |
| 目标商品认证编号标识 | tapproval\_id | 目标商品认证编号标识 | bigint(20) |
| 平台商品认证编号标识 | papproval\_id | 平台商品认证编号标识 | bigint(20) |
| 匹配标志 | ship\_flag | 0 未匹配 1已匹配 | int(1) |
| 审核状态 | check\_status | 审核状态 | int(4) |
| 删除标志 | del\_flag | 0否 1是 | int(1) |
| 创建人标识 | cremanid | 创建人标识 | bigint(20) |
| 创建人 | cremanname | 创建人 | varchar(30) |
| 创建时间 | credate | 创建时间 | datetime |
| 修改人标识 | editmanid | 修改人标识 | bigint(20) |
| 修改人 | editmanname | 修改人 | varchar(30) |
| 修改时间 | editdate | 修改时间 | datetime |
| 备注 | memo | 备注 | varchar(200) |

#### 4.2.2.3表hdi\_unicode\_goods\_ship\_hist 商品匹配历史

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| 匹配历史标识 | shiphist\_id | 关系标识 | bigint(20) |
| 机构标识 | dept\_id | 机构标识 | bigint(20) |
| 目标机构标识 | torg\_id | 目标机构标识 | bigint(20) |
| 目标机构类型 | torg\_type | 目标机构类型 | int(4) |
| 目标商品类型 | tgoods\_type | 1 药品 2 试剂 3 耗材 | int(4) |
| 目标商品标识 | tgoods\_id | 目标商品标识 | bigint(20) |
| 平台商品标识 | pgoods\_id | 平台商品标识 | bigint(20) |
| 目标商品规格标识 | tspecs\_id | 目标商品规格标识 | bigint(20) |
| 平台商品规格标识 | pspecs\_id | 平台商品规格标识 | bigint(20) |
| 目标商品认证编号标识 | tapproval\_id | 目标商品认证编号标识 | bigint(20) |
| 平台商品认证编号标识 | papproval\_id | 平台商品认证编号标识 | bigint(20) |
| 匹配标志 | ship\_flag | 0 未匹配 1已匹配 | int(1) |
| 失败原因 | fail\_reason | 失败原因 | varchar(2000) |
| 审核状态 | check\_status | 审核状态 | int(4) |
| 删除标志 | del\_flag | 0否 1是 | int(1) |
| 创建人标识 | cremanid | 创建人标识 | bigint(20) |
| 创建人 | cremanname | 创建人 | varchar(30) |
| 创建时间 | credate | 创建时间 | datetime |
| 修改人标识 | editmanid | 修改人标识 | bigint(20) |
| 修改人 | editmanname | 修改人 | varchar(30) |
| 修改时间 | editdate | 修改时间 | datetime |
| 备注 | memo | 备注 | varchar(200) |

#### 4.2.2.4表hdi\_unicode\_consumables\_cate 耗材商品分类

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| 商品分类标识 | cate\_id | 商品分类标识 | bigint(20) |
| 分类名称 | cate\_name | 分类名称 | varchar(30) |
| 分类编码 | cate\_no | 分类编码 | varchar(30) |
| 分类层级 | cate\_level | 根节点为1，第一层，依次往下 | int(4) |
| 父分类标识 | pcate\_id | 父分类标识 | bigint(20) |
| 机构标识 | dept\_id | 机构标识 | bigint(20) |
| 备注 | memo | 备注 | varchar(200) |

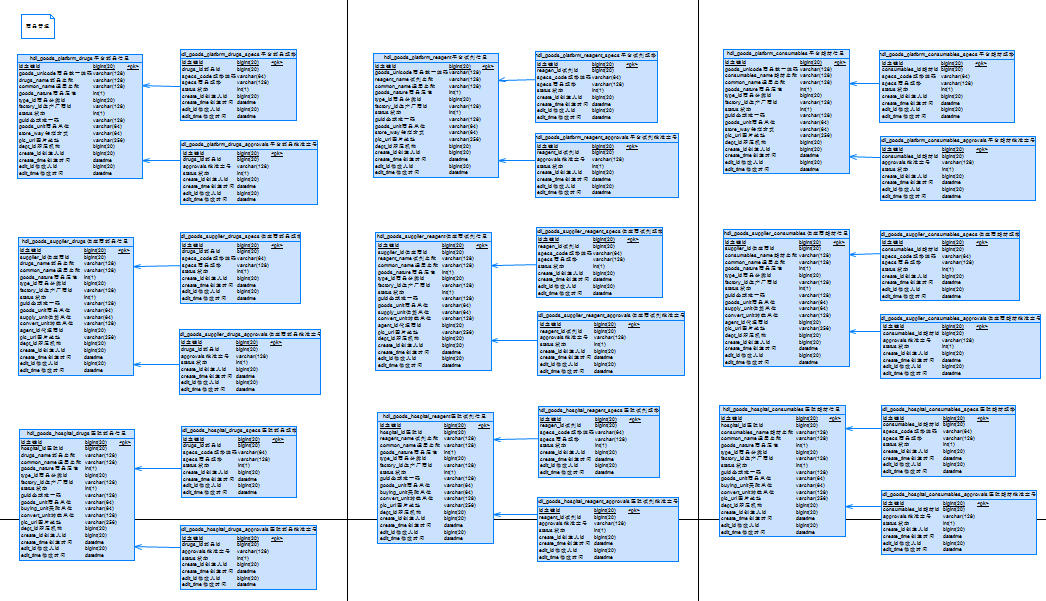
#### 4.2.2.5表hdi\_unicode\_reagent\_cate 试剂商品分类

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| 商品分类标识 | cate\_id | 商品分类标识 | bigint(20) |
| 分类名称 | cate\_name | 分类名称 | varchar(30) |
| 分类编码 | cate\_no | 分类编码 | varchar(30) |
| 分类层级 | cate\_level | 根节点为1，第一层，依次往下 | int(4) |
| 父分类标识 | pcate\_id | 父分类标识 | bigint(20) |
| 机构标识 | dept\_id | 机构标识 | bigint(20) |
| 备注 | memo | 备注 | varchar(200) |

#### 4.2.2.6表hdi\_unicode\_drugs\_cate 药品商品分类

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| 商品分类标识 | cate\_id | 商品分类标识 | bigint(20) |
| 分类名称 | cate\_name | 分类名称 | varchar(30) |
| 分类编码 | cate\_no | 分类编码 | varchar(30) |
| 分类层级 | cate\_level | 根节点为1，第一层，依次往下 | int(4) |
| 父分类标识 | pcate\_id | 父分类标识 | bigint(20) |
| 机构标识 | dept\_id | 机构标识 | bigint(20) |
| 备注 | memo | 备注 | varchar(200) |

## 4.3商品管理



### 4.3.1表清单

|  |  |
| --- | --- |
| 名称 | 说明 |
| hdi\_goods\_platform\_drugs | 平台药品信息 |
| hdi\_goods\_platform\_drugs\_specs | 平台药品规格 |
| hdi\_goods\_platform\_drugs\_approvals | 平台药品批准文号 |
| hdi\_goods\_supplier\_drugs | 供应商药品信息 |
| hdi\_goods\_supplier\_drugs\_specs | 供应商药品规格 |
| hdi\_goods\_supplier\_drugs\_approvals | 供应商药品批准文号 |
| hdi\_goods\_hospital\_drugs | 医院药品信息 |
| hdi\_goods\_hospital\_drugs\_specs | 医院药品规格 |
| hdi\_goods\_hospital\_drugs\_approvals | 医院药品批准文号 |
| hdi\_goods\_platform\_reagent | 平台试剂信息 |
| hdi\_goods\_platform\_reagent\_specs | 平台试剂规格 |
| hdi\_goods\_platform\_reagent\_approvals | 平台试剂批准文号 |
| hdi\_goods\_supplier\_reagent | 供应商试剂信息 |
| hdi\_goods\_supplier\_reagent\_specs | 供应商试剂规格 |
| hdi\_goods\_supplier\_reagent\_approvals | 供应商试剂批准文号 |
| hdi\_goods\_hospital\_reagent | 医院试剂信息 |
| hdi\_goods\_hospital\_reagent\_specs | 医院试剂规格 |
| hdi\_goods\_hospital\_reagent\_approvals | 医院试剂批准文号 |
| hdi\_goods\_platform\_consumables | 平台耗材信息 |
| hdi\_goods\_platform\_consumables\_specs | 平台耗材规格 |
| hdi\_goods\_platform\_consumables\_approvals | 平台耗材批准文号 |
| hdi\_goods\_supplier\_consumables | 供应商耗材信息 |
| hdi\_goods\_supplier\_consumables\_specs | 供应商耗材规格 |
| hdi\_goods\_supplier\_consumables\_approvals | 供应商耗材批准文号 |
| hdi\_goods\_hospital\_consumables | 医院耗材信息 |
| hdi\_goods\_hospital\_consumables\_specs | 医院耗材规格 |
| hdi\_goods\_hospital\_consumables\_approvals | 医院耗材批准文号 |

### 4.3.2表结构

#### 4.3.2.1表hdi\_goods\_platform\_drugs 平台药品信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| goods\_unicode 商品统一编码 | goods\_unicode | 商品统一编码 | varchar(128) |
| drugs\_name 药品名称 | drugs\_name | 药品名称 | varchar(128) |
| common\_name 通用名称 | common\_name | 通用名称 | varchar(128) |
| goods\_nature 商品属性 | goods\_nature | 商品属性(0:国产;1:进口) | int(1) |
| type\_id 商品分类id | type\_id | 商品分类id | bigint(20) |
| factory\_id 生产厂商id | factory\_id | 生产厂商id | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| goods\_unit 商品单位 | goods\_unit | 商品单位 | varchar(64) |
| store\_way 储存方式 | store\_way | 储存方式 | varchar(64) |
| pic\_url 图片地址 | pic\_url | 图片地址 | varchar(256) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.2表hdi\_goods\_platform\_drugs\_specs 平台药品规格

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| drugs\_id 药品id | drugs\_id | 药品id | bigint(20) |
| specs\_code 规格编码 | specs\_code | 规格编码 | varchar(64) |
| specs 商品规格 | specs | 商品规格 | varchar(128) |
| guid 全球唯一码 | guid | 全球唯一码 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.3表hdi\_goods\_platform\_drugs\_approvals 平台药品批准文号

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| drugs\_id 药品id | drugs\_id | 药品id | bigint(20) |
| approvals 批准文号 | approvals | 批准文号 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.4表hdi\_goods\_supplier\_drugs 供应商药品信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| supplier\_id 供应商id | supplier\_id | 供应商id | bigint(20) |
| drugs\_name 药品名称 | drugs\_name | 药品名称 | varchar(128) |
| common\_name 通用名称 | common\_name | 通用名称 | varchar(128) |
| goods\_nature 商品属性 | goods\_nature | 商品属性(0:国产;1:进口) | int(1) |
| type\_id 商品分类id | type\_id | 商品分类id | bigint(20) |
| factory\_id 生产厂商id | factory\_id | 生产厂商id | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| goods\_unit 商品单位 | goods\_unit | 商品单位 | varchar(64) |
| supply\_unit 供货单位 | supply\_unit | 供货单位 | varchar(64) |
| convert\_unit 转换单位 | convert\_unit | 转换单位 | varchar(128) |
| agent\_id 代理商id | agent\_id | 代理商id | bigint(20) |
| pic\_url 图片地址 | pic\_url | 图片地址 | varchar(256) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.5表hdi\_goods\_supplier\_drugs\_specs 供应商药品规格

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| drugs\_id 药品id | drugs\_id | 药品id | bigint(20) |
| specs\_code 规格编码 | specs\_code | 规格编码 | varchar(64) |
| specs 商品规格 | specs | 商品规格 | varchar(128) |
| guid 全球唯一码 | guid | 全球唯一码 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.6表hdi\_goods\_supplier\_drugs\_approvals 供应商药品批准文号

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| drugs\_id 药品id | drugs\_id | 药品id | bigint(20) |
| approvals 批准文号 | approvals | 批准文号 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.7表hdi\_goods\_hospital\_drugs 医院药品信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| hospital\_id 医院id | hospital\_id | 医院id | bigint(20) |
| drugs\_name 药品名称 | drugs\_name | 药品名称 | varchar(128) |
| common\_name 通用名称 | common\_name | 通用名称 | varchar(128) |
| goods\_nature 商品属性 | goods\_nature | 商品属性(0:国产;1:进口) | int(1) |
| type\_id 商品分类id | type\_id | 商品分类id | bigint(20) |
| factory\_id 生产厂商id | factory\_id | 生产厂商id | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| goods\_unit 商品单位 | goods\_unit | 商品单位 | varchar(64) |
| buying\_unit 采购单位 | buying\_unit | 采购单位 | varchar(64) |
| convert\_unit 转换单位 | convert\_unit | 转换单位 | varchar(128) |
| pic\_url 图片地址 | pic\_url | 图片地址 | varchar(256) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.8表hdi\_goods\_hospital\_drugs\_specs 医院药品规格

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| drugs\_id 药品id | drugs\_id | 药品id | bigint(20) |
| specs\_code 规格编码 | specs\_code | 规格编码 | varchar(64) |
| specs 商品规格 | specs | 商品规格 | varchar(128) |
| guid 全球唯一码 | guid | 全球唯一码 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.9表hdi\_goods\_hospital\_drugs\_approvals 医院药品批准文号

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| drugs\_id 药品id | drugs\_id | 药品id | bigint(20) |
| approvals 批准文号 | approvals | 批准文号 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.10表hdi\_goods\_platform\_reagent 平台试剂信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| goods\_unicode 商品统一编码 | goods\_unicode | 商品统一编码 | varchar(128) |
| reagent\_name 试剂名称 | reagent\_name | 试剂名称 | varchar(128) |
| common\_name 通用名称 | common\_name | 通用名称 | varchar(128) |
| goods\_nature 商品属性 | goods\_nature | 商品属性(0:国产;1:进口) | int(1) |
| type\_id 商品分类id | type\_id | 商品分类id | bigint(20) |
| factory\_id 生产厂商id | factory\_id | 生产厂商id | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| goods\_unit 商品单位 | goods\_unit | 商品单位 | varchar(64) |
| store\_way 储存方式 | store\_way | 储存方式 | varchar(64) |
| pic\_url 图片地址 | pic\_url | 图片地址 | varchar(256) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.11表hdi\_goods\_platform\_reagent\_specs 平台试剂规格

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| reagen\_id 试剂id | reagen\_id | 试剂id | bigint(20) |
| specs\_code 规格编码 | specs\_code | 规格编码 | varchar(64) |
| specs 商品规格 | specs | 商品规格 | varchar(128) |
| guid 全球唯一码 | guid | 全球唯一码 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.12表hdi\_goods\_platform\_reagent\_approvals 平台试剂批准文号

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| reagent\_id 试剂id | reagent\_id | 试剂id | bigint(20) |
| approvals 批准文号 | approvals | 批准文号 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.13表hdi\_goods\_supplier\_reagent 供应商试剂信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| supplier\_id 供应商id | supplier\_id | 供应商id | bigint(20) |
| reagent\_name 试剂名称 | reagent\_name | 试剂名称 | varchar(128) |
| common\_name 通用名称 | common\_name | 通用名称 | varchar(128) |
| goods\_nature 商品属性 | goods\_nature | 商品属性(0:国产;1:进口) | int(1) |
| type\_id 商品分类id | type\_id | 商品分类id | bigint(20) |
| factory\_id 生产厂商id | factory\_id | 生产厂商id | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| goods\_unit 商品单位 | goods\_unit | 商品单位 | varchar(64) |
| supply\_unit 供货单位 | supply\_unit | 供货单位 | varchar(64) |
| convert\_unit 转换单位 | convert\_unit | 转换单位 | varchar(128) |
| agent\_id 代理商id | agent\_id | 代理商id | bigint(20) |
| pic\_url 图片地址 | pic\_url | 图片地址 | varchar(256) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.14表hdi\_goods\_supplier\_reagent\_specs 供应商试剂规格

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| reagen\_id 试剂id | reagen\_id | 试剂id | bigint(20) |
| specs\_code 规格编码 | specs\_code | 规格编码 | varchar(64) |
| specs 商品规格 | specs | 商品规格 | varchar(128) |
| guid 全球唯一码 | guid | 全球唯一码 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.15表hdi\_goods\_supplier\_reagent\_approvals 供应商试剂批准文号

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| reagent\_id 试剂id | reagent\_id | 试剂id | bigint(20) |
| approvals 批准文号 | approvals | 批准文号 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.16表hdi\_goods\_hospital\_reagent 医院试剂信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| hospital\_id 医院id | hospital\_id | 医院id | bigint(20) |
| reagent\_name 试剂名称 | reagent\_name | 试剂名称 | varchar(128) |
| common\_name 通用名称 | common\_name | 通用名称 | varchar(128) |
| goods\_nature 商品属性 | goods\_nature | 商品属性(0:国产;1:进口) | int(1) |
| type\_id 商品分类id | type\_id | 商品分类id | bigint(20) |
| factory\_id 生产厂商id | factory\_id | 生产厂商id | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| goods\_unit 商品单位 | goods\_unit | 商品单位 | varchar(64) |
| buying\_unit 采购单位 | buying\_unit | 采购单位 | varchar(64) |
| convert\_unit 转换单位 | convert\_unit | 转换单位 | varchar(128) |
| pic\_url 图片地址 | pic\_url | 图片地址 | varchar(256) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.17表hdi\_goods\_hospital\_reagent\_specs 医院试剂规格

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| reagen\_id 试剂id | reagen\_id | 试剂id | bigint(20) |
| specs\_code 规格编码 | specs\_code | 规格编码 | varchar(64) |
| specs 商品规格 | specs | 商品规格 | varchar(128) |
| guid 全球唯一码 | guid | 全球唯一码 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.18表hdi\_goods\_hospital\_reagent\_approvals 医院试剂批准文号

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| reagent\_id 试剂id | reagent\_id | 试剂id | bigint(20) |
| approvals 批准文号 | approvals | 批准文号 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.19表hdi\_goods\_platform\_consumables 平台耗材信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| goods\_unicode 商品统一编码 | goods\_unicode | 商品统一编码 | varchar(128) |
| consumables\_name 耗材名称 | consumables\_name | 耗材名称 | varchar(128) |
| common\_name 通用名称 | common\_name | 通用名称 | varchar(128) |
| goods\_nature 商品属性 | goods\_nature | 商品属性(0:国产;1:进口) | int(1) |
| type\_id 商品分类id | type\_id | 商品分类id | bigint(20) |
| factory\_id 生产厂商id | factory\_id | 生产厂商id | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| goods\_unit 商品单位 | goods\_unit | 商品单位 | varchar(64) |
| store\_way 储存方式 | store\_way | 储存方式 | varchar(64) |
| pic\_url 图片地址 | pic\_url | 图片地址 | varchar(256) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.20表hdi\_goods\_platform\_consumables\_specs 平台耗材规格

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| consumables\_id 耗材id | consumables\_id | 耗材id | bigint(20) |
| specs\_code 规格编码 | specs\_code | 规格编码 | varchar(64) |
| specs 商品规格 | specs | 商品规格 | varchar(128) |
| guid 全球唯一码 | guid | 全球唯一码 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.21表hdi\_goods\_platform\_consumables\_approvals 平台耗材批准文号

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| consumables\_id 耗材id | consumables\_id | 耗材id | bigint(20) |
| approvals 批准文号 | approvals | 批准文号 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.22表hdi\_goods\_supplier\_consumables 供应商耗材信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| supplier\_id 供应商id | supplier\_id | 供应商id | bigint(20) |
| consumables\_name 耗材名称 | consumables\_name | 耗材名称 | varchar(128) |
| common\_name 通用名称 | common\_name | 通用名称 | varchar(128) |
| goods\_nature 商品属性 | goods\_nature | 商品属性(0:国产;1:进口) | int(1) |
| type\_id 商品分类id | type\_id | 商品分类id | bigint(20) |
| factory\_id 生产厂商id | factory\_id | 生产厂商id | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| goods\_unit 商品单位 | goods\_unit | 商品单位 | varchar(64) |
| supply\_unit 供货单位 | supply\_unit | 供货单位 | varchar(64) |
| convert\_unit 转换单位 | convert\_unit | 转换单位 | varchar(128) |
| agent\_id 代理商id | agent\_id | 代理商id | bigint(20) |
| pic\_url 图片地址 | pic\_url | 图片地址 | varchar(256) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.23表hdi\_goods\_supplier\_consumables\_specs 供应商耗材规格

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| consumables\_id 耗材id | consumables\_id | 耗材id | bigint(20) |
| specs\_code 规格编码 | specs\_code | 规格编码 | varchar(64) |
| specs 商品规格 | specs | 商品规格 | varchar(128) |
| guid 全球唯一码 | guid | 全球唯一码 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.24表hdi\_goods\_supplier\_consumables\_approvals 供应商耗材批准文号

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| consumables\_id 耗材id | consumables\_id | 耗材id | bigint(20) |
| approvals 批准文号 | approvals | 批准文号 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.25表hdi\_goods\_hospital\_consumables 医院耗材信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| hospital\_id 医院id | hospital\_id | 医院id | bigint(20) |
| consumables\_name 耗材名称 | consumables\_name | 耗材名称 | varchar(128) |
| common\_name 通用名称 | common\_name | 通用名称 | varchar(128) |
| goods\_nature 商品属性 | goods\_nature | 商品属性(0:国产;1:进口) | int(1) |
| type\_id 商品分类id | type\_id | 商品分类id | bigint(20) |
| factory\_id 生产厂商id | factory\_id | 生产厂商id | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| goods\_unit 商品单位 | goods\_unit | 商品单位 | varchar(64) |
| buying\_unit 采购单位 | buying\_unit | 采购单位 | varchar(64) |
| convert\_unit 转换单位 | convert\_unit | 转换单位 | varchar(128) |
| pic\_url 图片地址 | pic\_url | 图片地址 | varchar(256) |
| dept\_id 所属机构 | dept\_id | 所属机构 | bigint(20) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.26表hdi\_goods\_hospital\_consumables\_specs 医院耗材规格

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| consumables\_id 耗材id | consumables\_id | 耗材id | bigint(20) |
| specs\_code 规格编码 | specs\_code | 规格编码 | varchar(64) |
| specs 商品规格 | specs | 商品规格 | varchar(128) |
| guid 全球唯一码 | guid | 全球唯一码 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

#### 4.3.2.27表hdi\_goods\_hospital\_consumables\_approvals 医院耗材批准文号

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| consumables\_id 耗材id | consumables\_id | 耗材id | bigint(20) |
| approvals 批准文号 | approvals | 批准文号 | varchar(128) |
| status 状态 | status | 状态(0:停用;1:启用) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |

## 4.4证照管理

### 4.4.1表清单

### 4.4.2表结构

## 4.5采购管理

### 4.5.1表清单

### 4.5.2表结构

## 4.6供货管理

### 4.6.1表清单

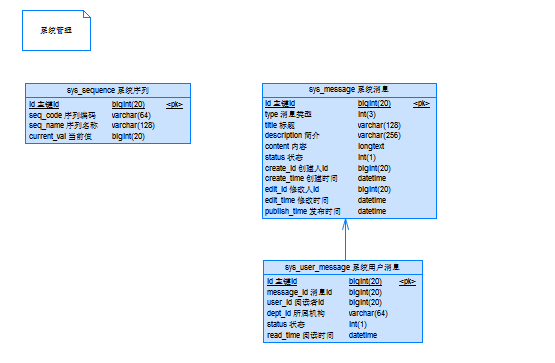
### 4.6.2表结构

## 4.7退货管理

### 4.7.1表清单

### 4.7.2表结构

## 4.8系统消息



### 4.8.1表清单

|  |  |
| --- | --- |
| 名称 | 说明 |
| sys\_message | 系统消息 |
| sys\_user\_message | 系统用户消息 |

### 4.8.2表结构

#### 4.8.2.1表sys\_message 系统消息的栏的清单

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| type 消息类型 | type | 消息类型(0:系统通知;1:证照提醒) | int(3) |
| title 标题 | title | 标题 | varchar(128) |
| description 简介 | description | 简介 | varchar(256) |
| content 内容 | content | 内容 | longtext |
| status 状态 | status | 状态(0:草稿;1:已发布;2:已删除) | int(1) |
| create\_id 创建人id | create\_id | 创建人id | bigint(20) |
| create\_time 创建时间 | create\_time | 创建时间 | datetime |
| edit\_id 修改人id | edit\_id | 修改人id | bigint(20) |
| edit\_time 修改时间 | edit\_time | 修改时间 | datetime |
| publish\_time 发布时间 | publish\_time | 发布时间 | datetime |

#### 4.8.2.2表sys\_user\_message 系统用户消息的栏的清单

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 代码 | 注释 | 数据类型 |
| id 主键id | id | 主键id | bigint(20) |
| message\_id 消息id | message\_id | 消息id | bigint(20) |
| user\_id 阅读者id | user\_id | 阅读者id | bigint(20) |
| dept\_id 所属机构 | dept\_id | 所属机构 | varchar(64) |
| status 状态 | status | 状态(0:未读;1:已读;2:删除) | int(1) |
| read\_time 阅读时间 | read\_time | 阅读时间 | datetime |

# 模块设计

## 5.1机构管理

### 5.1.1供应商信息

#### 5.1.1.1用例图



#### 5.1.1.2功能设计说明

##### 5.1.1.2.1分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 查询条件 | 机构名称、机构所属区域、状态 |
| 列表显示 | 1. 显示字段：机构编码、机构名称、机构所属区域、是否集团机构、状态 2. 排序：先按修改时间，再按创建时间倒序 |
| 类包 | com.ebig.hdi.modules.org |
| 输入 | {  "page":"1",  "limit":"10",  "supplierName":"...医药有限公司",  ...  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  String supplierName = (String)params.get("supplierName");  String provinceCode = (String)params.get("provinceCode");  String cityCode = (String)params.get("cityCode");  String areaCode = (String)params.get("areaCode");    Page<OrgSupplierInfoEntity> page = this.selectPage(  new Query<OrgSupplierInfoEntity>(params).getPage(),  new EntityWrapper<OrgSupplierInfoEntity>()  .addFilterIfNeed(params.get(Constant.SQL\_FILTER) != null, (String)params.get(Constant.SQL\_FILTER))  .like(StringUtils.isNotBlank(supplierName), "supplier\_name", supplierName)  .eq(StringUtils.isNotBlank(provinceCode), "province\_code", provinceCode)  .eq(StringUtils.isNotBlank(cityCode), "city\_code", cityCode)  .eq(StringUtils.isNotBlank(areaCode), "area\_code", areaCode)  .eq(params.get("status")!=null, "status", params.get("status"))  );  return new PageUtils(page);  } |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": 1,  "supplierCode": "440980120001",  "supplierName": "广州白云山医药有限公司",  "creditCode": "0209010098734",  "provinceCode": "44",  "cityCode": "4401",  "areaCode": "440106",  ......  },  {  "id": 2,  ......  }  ]  }  } |

##### 5.1.1.2.2新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1. 机构编码（系统按“机构所属区域+自编码”规则自动生成唯一编码） 2. 机构名称（要求是营业执照上的公司名称），唯一性约束 3. 机构信用代码（机构的统一社会信息代码），唯一性约束 4. 保存供应商信息时，系统自动生成一个该供应商对应的管理账号，供应商管理员可通过管理账号登录系统进行权限范围内的管理操作。 |
| 必填字段 | 机构名称、机构信用代码、机构所属区域、状态 |
| 选填字段 | 机构性质，法人、企业地址、联系电话、邮箱、传真 |
| 类包 | com.ebig.hdi.modules.org |
| 输入 | {  "supplierName":"机构名称",  "creditCode":"机构信用代码",  "provinceCode": "44",  "cityCode": "4401",  "areaCode": "440106",  "status": "1",  ...  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void insert(OrgSupplierInfoEntity orgSupplierInfoEntity) {  //生成机构编码  Long supplierCode = ......;  orgSupplierInfoEntity.setSupplierCode(supplierCode);  orgSupplierInfoEntity.setCreateId(getUserId());  orgSupplierInfoEntity.setCreateTime(new Date());  this.insert(orgSupplierInfoEntity);    //创建系统机构信息    //创建供应商角色信息    //分配供应商菜单、数据权限    //创建供应商管理员    } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.1.1.2.3编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、机构名称（要求是营业执照上的公司名称），唯一性约束  2、机构信用代码（机构的统一社会信息代码），唯一性约束 |
| 必填字段 | 机构名称、机构信用代码、机构所属区域、状态 |
| 选填字段 | 机构性质，法人、企业地址、联系电话、邮箱、传真 |
| 类包 | com.ebig.hdi.modules.org |
| 输入 | {  "id":"机构id",  "supplierName":"机构名称",  "creditCode":"机构信用代码",  "provinceCode": "44",  "cityCode": "4401",  "areaCode": "440106",  "status": "1",  ...  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void update(OrgSupplierInfoEntity orgSupplierInfoEntity) {  //更新供应商信息  orgSupplierInfoEntity.setEditId(getUserId());  orgSupplierInfoEntity.setEditTime(new Date());  orgSupplierInfoService.updateById(orgSupplierInfoEntity);  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.1.1.2.4查看详情

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 显示字段 | 机构名称、机构信用代码、机构所属区域、状态、是否集团机构、机构编码、机构性质、法人、企业地址、联系电话、邮箱、传真 |
| 类包 | com.ebig.hdi.modules.org |
| 输入 | {  "id":"机构id"  } |
| 业务代码 | @Override  public OrgSupplierInfoEntity selectById(Long id) {  return this.baseMapper.selectById(id);  } |
| 输出 | {  "msg": "success",  "code": 0,  "orgSupplierInfoEntity ": {  "id": 1,  "supplierCode": "440980120001",  "supplierName": "广州白云山医药有限公司",  "creditCode": "0209010098734",  "provinceCode": "44",  "cityCode": "4401",  "areaCode": "440106",  ......  }  } |

##### 5.1.1.2.5绑定医院

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 需求 | 设置供应商和医院的绑定关系，“绑定医院”界面自动显示平台统一维护可用的医院信息，一个供应商可以绑定一个或者多个医院。 |
| 医院分页查询接口 | 可查询医院信息分页查询设计相关内容 |
| 类包 | com.ebig.hdi.modules.org |
| 输入 | {  "supplierCode": "440980120001",  "hospitalCode": "440980120001,440980120002,440980120003"  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void insert(OrgSupplierHospitalRefEntity orgSupplierHospitalRefEntity) {  //保存供应商医院绑定关系  Strring[] hospitalCodeArray = orgSupplierHospitalRefEntity.getHospitalCode();  for(int i=0; i < hospitalCodeArray.length; i++){  orgSupplierHospitalRefEntity.setHospitalCode(hospitalCodeArray[i]);  orgSupplierHospitalRefEntity.setCreateId(getUserId());  orgSupplierHospitalRefEntity.setCreateTime(new Date());  this.insert(orgSupplierHospitalRefEntity);  }  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.1.1.2.6集团授权

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 需求 | 1、平台管理员可以对机构进行集团授权，授权时可限制集团机构可添加的下级机构数量。  2、已授权的集团机构管理员登录系统，可以查看“集团管理”模块并进行权限范围内的操作。 |
| 必填字段 | 是否集团机构、集团机构数据 |
| 选填字段 | 上层父级 |
| 选择上层父级接口 | 可查看供应商集团信息列表查询设计相关内容 |
| 输入 | {  "id": "1",  "isGroup": "1",  "childNumber": "5",  "parentCode": ""  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void insert(Map<String, Object> params) {  //更新供应商信息  OrgSupplierInfoEntity orgSupplierInfoEntity = new OrgSupplierInfoEntity();  orgSupplierInfoEntity.setId(params.get("id"));  orgSupplierInfoEntity.setIsGroup(params.get("isGroup"));  orgSupplierInfoEntity.setChildNumber(params.get("childNumber"));  orgSupplierInfoService.updateById(orgSupplierInfoEntity);  //插入供应商机构信息  OrgSupplierGroupEntity orgSupplierGroupEntity = new OrgSupplierGroupEntity();  orgSupplierGroupEntity.set....  orgSupplierGroupEntity.set....  orgSupplierGroupEntity.set....  ......  orgSupplierGroupEntity.insert(orgSupplierGroupEntity);  } |
| 输出 | {  "msg": "success",  "code": 0  } |

### 5.1.2医院信息

#### 5.1.2.1用例图



#### 5.1.2.2功能设计说明

### 5.1.3厂商信息（平台）

#### 5.1.3.1用例图



#### 5.1.3.2功能设计说明

##### 5.1.3.2.1分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 查询条件 | 厂商名称、厂商所属区域、状态 |
| 列表显示 | 厂商编码、厂商名称、厂商信用代码、厂商所属区域、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 输入 | {  "page":"1",  "limit":"10",  "factoryName":"",  "countryCode":"",  "provinceCode":"44",  "cityCode":"",  "status":"1"  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  Page<OrgFactoryInfoEntity> page = this.selectPage(  new Query<OrgFactoryInfoEntity>(params).getPage(),  new EntityWrapper<OrgFactoryInfoEntity>()  .addFilterIfNeed(params.get(Constant.SQL\_FILTER) != null, (String)params.get(Constant.SQL\_FILTER))  .like(StringUtils.isNotBlank(params.get("factoryName").toString()), "factory\_name", params.get("factoryName").toString())  .eq(StringUtils.isNotBlank(params.get("countryCode").toString()), "country\_code", params.get("countryCode").toString())  .eq(StringUtils.isNotBlank(params.get("provinceCode").toString()), "province\_code", params.get("provinceCode").toString())  .eq(StringUtils.isNotBlank(params.get("cityCode").toString()), "city\_code", params.get("cityCode").toString())  .eq(StringUtils.isNotBlank(params.get("areaCode").toString()), "area\_code", params.get("areaCode").toString())  .eq(StringUtils.isNotBlank(params.get("status").toString()), "status", params.get("status").toString())  );  return new PageUtils(page);  } |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": "1",  "factoryCode": 1223,  "factoryName": "广州白云山医药有限公司",  "creditCode": "GZ0009381784",  "countryCode": "11",  "provinceCode": "44",  "cityCode": "44020",  "areaCode": "44020890",  "status": "1",  ......  },  {  "id": "2",  "factoryCode": 3456,  "factoryName": "广州白云山中一药业有限公司",  "creditCode": "GZ0009389834",  "countryCode": "11",  "provinceCode": "44",  "cityCode": "44020",  "areaCode": "44020890",  "status": "1",  ......  }......  ]  }  } |

##### 5.1.3.2.2新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、厂商编码（系统按“所属区域+自编码”规则自动生成唯一编码）  2、厂商名称，唯一性约束  3、厂商信用代码，唯一性约束 |
| 必填字段 | 厂商名称、厂商信用代码、厂商所属区域、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 业务代码 | {  "factoryName":"广州白云山制药总厂",  "countryCode":"11",  "provinceCode":"44",  "cityCode":"4409",  "status":"1"  } |
| 输出 | @Override  @Transactional(rollbackFor = Exception.class)  public void insert(OrgFactoryInfoEntity orgFactoryInfoEntity) {  //校验厂商名称、厂商信用代码  ......  //生成厂商编码  Long factoryCode= ......;  orgFactoryInfoEntity.setFactoryCode(factoryCode);  orgFactoryInfoEntity.setCreateId(getUserId());  orgFactoryInfoEntity.setCreateTime(new Date());  ......  this.insert(orgSupplierGroupEntity);  } |
| 业务代码 | {  "msg": "success",  "code": 0  } |

##### 5.1.3.2.3编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、厂商名称，唯一性约束  2、厂商信用代码，唯一性约束 |
| 必填字段 | 厂商名称、厂商信用代码、厂商所属区域、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 业务代码 | {  "id":"1",  "factoryName":"广州白云山制药总厂",  "countryCode":"11",  "provinceCode":"44",  "cityCode":"4409",  "status":"1"  } |
| 输出 | @Override  @Transactional(rollbackFor = Exception.class)  public void update(OrgFactoryInfoEntity orgFactoryInfoEntity) {  //校验厂商名称、厂商信用代码  ......  orgFactoryInfoEntity.setCreateId(getUserId());  orgFactoryInfoEntity.setCreateTime(new Date());  ......  this.updateById(orgSupplierGroupEntity);  } |
| 业务代码 | {  "msg": "success",  "code": 0  } |

##### 5.1.3.2.4查看详情

|  |  |
| --- | --- |
| 事项 | 内容 |
| 列表显示 | 厂商编码、厂商名称、厂商信用代码、厂商所属区域、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 输入 | http://localhost:2320/hdi-server-admin/org/orgfactoryinfo/info/1 |
| 业务代码 | @RequestMapping("/info/{id}")  public Hdi info(@PathVariable("id") Long id){  OrgFactoryInfoEntity orgFactoryInfo = orgFactoryInfoService.selectById(id);  return Hdi.ok().put("orgFactoryInfo", orgFactoryInfo);  } |
| 输出 | {  "msg": "success",  "code": 0,  "orgFactoryInfo": {  "id": "1",  "factory\_code": 1223,  "factory\_name": "广州白云山医药有限公司",  "creditCode": "GZ0009381784",  "country\_code": "11",  "provinceCode": "44",  "cityCode": "44020",  "areaCode": "44020890",  "status": "1",  ......  },  } |

##### 5.1.3.2.4删除

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.org |
| 功能说明 | 做软删除，更新状态操作 |
| 输入 | [1,2,3] |
| 业务代码 | @Override  public void delete(Long[] ids) {  for(Long id : ids) {  OrgFactoryInfoEntity orgFactoryInfoEntity= new OrgFactoryInfoEntity();  orgFactoryInfoEntity.setId(id);  //设置信息为删除状态  orgFactoryInfoEntity.set(StatusEnum.DELETE.getKey());  this.updateById(orgAgentInfoEntity);  }  } |
| 输出 | {  "msg": "success",  "code": 0  } |

### 5.1.4代理商信息

#### 5.1.4.1用例图



#### 5.1.4.2功能设计说明

##### 5.1.4.2.1分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 查询条件 | 机构名称、状态 |
| 列表显示 | 1. 显示字段：机构编码、机构名称、状态、证照状态   2、证照状态（判断该代理商证照是否上传，如果已上传，统计显示代理商的证照数量，点击文字链接直接跳转到“代理商证照管理”页面）   1. 排序：先按修改时间，再按创建时间倒序 |
| 类包 | com.ebig.hdi.modules.org |
| 输入 | {  "page":1,  "limit":10,  "agentName":"...医药有限公司",  "status":1,  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  Page<OrgSupplierInfoEntity> page = this.selectPage(  new Query<OrgSupplierInfoEntity>(params).getPage(),  new EntityWrapper<OrgSupplierInfoEntity>()  .addFilterIfNeed(params.get(Constant.SQL\_FILTER) != null, (String)params.get(Constant.SQL\_FILTER))  .like(StringUtils.isNotBlank(params.get("agentName")), "agent\_name", params.get("agentName"))  .eq(params.get("status")!=null, "status", params.get("status"))  );  return new PageUtils(page);  } |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": 1,  "supplierId": 7,  "agentCode": 44011128,  "agentName": "测试代理商1",  "creditCode": "dls001",  "status": 1,  ......  },  {  "id": 2,  "supplierId": 7,  "agentCode": 44011130,  "agentName": "测试代理商2",  "creditCode": "dls002",  "status": 1,  ......  }  ]  }  } |

##### 5.1.4.2.2新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、代理商编码（系统按“所属区域+自编码”规则自动生成唯一编码）  2、代理商名称，唯一性约束  3、代理商信用代码，唯一性约束 |
| 必填字段 | 代理商名称、代理商信用代码、代理商所属区域、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 业务代码 | {  "agentName":"代理商测试",  "creditCode": "dls100",  "countryCode":"11",  "provinceCode":"44",  "cityCode":"4409",  "status":"1"  } |
| 输出 | @Override  @Transactional(rollbackFor = Exception.class)  public void save(OrgAgentInfoEntity orgAgentInfoEntity) {  List<OrgAgentInfoEntity> agentName = baseMapper.selectByAgentName(orgAgentInfoEntity.getAgentName());  if(!StringUtil.isEmpty(agentName)){  throw new HdiException("代理商名称已存在");  }  List<OrgAgentInfoEntity> creditCode = baseMapper.selectByCreditCode(orgAgentInfoEntity.getCreditCode());  if(!StringUtil.isEmpty(creditCode)){  throw new HdiException("代理商信用代码已存在");  }  //获取供应商编码序列号  long seqValue = sysSequenceService.selectSeqValueBySeqCode(SequenceEnum.AGENT\_CODE.getKey());  long agentCode = Long.valueOf(orgAgentInfoEntity.getAreaCode() + String.valueOf(seqValue));  orgAgentInfoEntity.setAgentCode(agentCode);  orgAgentInfoEntity.setCreateTime(new Date());  this.insert(orgAgentInfoEntity);  } |
| 业务代码 | {  "msg": "success",  "code": 0  } |

##### 5.1.4.2.3编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、代理商名称，唯一性约束  2、代理商信用代码，唯一性约束 |
| 必填字段 | 代理商名称、代理商信用代码、代理商所属区域、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 业务代码 | {  "id":"1",  "agentName":"代理商测试",  "creditCode": "dls100",  "countryCode":"11",  "provinceCode":"44",  "cityCode":"4409",  "status":"1"  ......  } |
| 输出 | @Override  @Transactional(rollbackFor = Exception.class)  public void update(OrgAgentInfoEntity orgAgentInfoEntity) {  Long id = orgAgentInfoEntity.getId();  List<OrgAgentInfoEntity> agentName = baseMapper.selectByAgentName(orgAgentInfoEntity.getAgentName());  if(!StringUtil.isEmpty(agentName)&&agentName.get(0).getId()!=id){  throw new HdiException("代理商名称已存在");  }  List<OrgAgentInfoEntity> creditCode = baseMapper.selectByCreditCode(orgAgentInfoEntity.getCreditCode());  if(!StringUtil.isEmpty(creditCode)&&creditCode.get(0).getId()!=id){  throw new HdiException("代理商信用代码已存在");  }  orgAgentInfoEntity.setEditTime(new Date());  this.updateById(orgAgentInfoEntity);  } |
| 业务代码 | {  "msg": "success",  "code": 0  } |

##### 5.1.4.2.4查看详情

|  |  |
| --- | --- |
| 事项 | 内容 |
| 列表显示 | 代理编码、代理名称、代理商信用代码、代理商所属区域、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 输入 | http://localhost:2320/hdi-server-admin/org/orgagentinfo/info/2 |
| 业务代码 | @RequestMapping("/info/{id}")  public Hdi info(@PathVariable("id") Long id){  OrgAgentInfoEntity orgAgentInfo = orgAgentInfoService.selectById(id);  return Hdi.ok().put("orgAgentInfo", orgAgentInfo);  } |
| 输出 | {  "msg": "success",  "code": 0,  "orgFactoryInfo": {  "id": 1,  "supplierId": 7,  "agentCode": 44011128,  "agentName": "测试代理商1",  "creditCode": "dls001",  "status": 1,  ......  }  } |

##### 5.1.4.2.5删除

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.org |
| 功能说明 | 做软删除，更新状态操作 |
| 输入 | [1,2,3] |
| 业务代码 | @Override  public void delete(Long[] ids) {  for(Long id : ids) {  OrgAgentInfoEntity orgAgentInfoEntity= new OrgAgentInfoEntity();  orgAgentInfoEntity.setId(id);  //设置信息为删除状态  orgAgentInfoEntity.setStatus(StatusEnum.DELETE.getKey());  this.updateById(orgAgentInfoEntity);  }  } |
| 输出 | {  "msg": "success",  "code": 0  } |

## 5.2统一编码管理

### 5.2.1商品分类管理

### 5.2.1.1查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 查询条件 | 类别名称 |
| 列表显示 | 1、后台字段：类别ID、类别名称、类别码、上层类别、层级、层级顺序  2、根节点ID默认为0  3、排序：按照后台提供的层级和层级顺序排序 |
| 类包 | com.ebig.hdi.modules.consumables  com.ebig.hdi.modules.drugs  com.ebig.hdi.modules.reagent |
| 输入 | {  “pcateId”:”0”  } |
| 业务代码 | **public** List<TreeNode> queryNode(Long pcateId){  List<Map<String, Object>> queryForList = jdbcTemplate.queryForList("select \* from hdi\_unicode\_consumables\_cate where pcate\_id=?", pcateId);  **for** (Map<String, Object> map : queryForList) {  TreeNode treeNode = **new** TreeNode();  //**TODO** 1.增加其他属性 2.叶子节点判断  treeNode.setChildren(queryNode((Long)map.get("cate\_id")));  nodeList.add(treeNode);  }  **return** nodeList;  } |
| 查询语句 | 查询对应插件的分类表（全查） |
| 输出 | **{**  **"msg"**: "success"**,**  **"code"**: 0**,**  **"data"**: **[**  **{**  **"cateId"**: 1**,**  **"cateName"**: "针剂"**,**  **"cateNo"**: "ISO8808-01"**,**  **"levelOrder"**: 1**,**  **"cateLevel"**: 1**,**  **"pcateId"**: 0**,**  **"deptId"**: 1,  **"leaf"**: false,  **"children":[…]**  **},**  **{**  **"cateId"**: 2**,**  **"cateName"**: "片剂"**,**  **"cateNo"**: "ISO8808-02"**,**  **"levelOrder"**: 2**,**  **"cateLevel"**: 1**,**  **"pcateId"**: 0**,**  **"deptId"**: 1,  **"leaf"**: false,  **"children":[…]**  **}**  **]**  **}** |
|  |  |

### 5.2.1.2新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 逻辑说明 | 1、前台传入字段：类别名称、类别码、上层类别ID、层级、层级顺序  2、同一层级，要求类别码不能重复  3、新增成功后返回新增的记录 |
| 类包 | com.ebig.hdi.modules.consumables  com.ebig.hdi.modules.drugs  com.ebig.hdi.modules.reagent |
| 输入 | {  "pcateId":0,  "cateName":"西药",  "cateNo":"ISO8808-01",  "cateLevel":1,  "levelOrder":1  } |
| 业务代码 | SQL如下：  select count(\*) from hdi\_unicode\_consumables\_cate where cate\_level=#{cateLevel} and cate\_no=#{cateNo}  若该值不为0,则不能新增 |
| 输出 | **{**  **"msg"**: "success"**,**  **"code"**: 0  **}** |
|  |  |

### 5.2.1.3编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 逻辑说明 | 1、前台传入字段：类别ID、类别名称、类别码、上层类别ID、层级、层级顺序  2、同一层级，要求类别码不能重复 |
| 类包 | com.ebig.hdi.modules.consumables  com.ebig.hdi.modules.drugs  com.ebig.hdi.modules.reagent |
| 输入 | {  "cateId":1,  "pcateId":0,  "cateName":"西药",  "cateNo":"ISO8808-01",  "cateLevel":1,  "levelOrder":1  } |
| 业务代码 | SQL如下：  select count(\*) from hdi\_unicode\_consumables\_cate where cate\_level=#{cateLevel} and cate\_no=#{cateNo}  若该值不为0,则不能编辑 |
| 输出 | **{**  **"msg"**: "success"**,**  **"code"**: 0  **}** |
|  |  |

### 5.2.1.4 删除

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 逻辑说明 | 1、前台传入字段：类别ID  2、树结构数据不要多选，删除时选中一节点，前端判断若有孩子，则询问是否级联删除所有孩子，传入类别ID、后台默认级联删除所有孩子。 |
| 类包 | com.ebig.hdi.modules.consumables  com.ebig.hdi.modules.drugs  com.ebig.hdi.modules.reagent |
| 输入 | {  "cateId":1  } |
| 业务代码 | **public** void deleteNode(Long cateId){  List<Map<String, Object>> queryForList = jdbcTemplate.queryForList("select \* from hdi\_unicode\_consumables\_cate where pcate\_id=?", cateId);  **for** (Map<String, Object> map : queryForList) {  //**TODO** 删除语句  deleteNode((Long)map.get("cate\_id")) }  //上面只是级联删除孩子，最后记得把自己也删了  } |
| 输出 | **{**  **"msg"**: "success"**,**  **"code"**: 0  **}** |
|  |  |

### 5.2.2供应商匹配管理

插件：hdi-server-admin

类包：com.ebig.hdi.modules.unicode

### 5.2.2.1 分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 查询条件 | 医院名称、医院供应商名称、匹配状态 |
| 列表显示 | 1. 后台返回字段：关系标识、医院名称、医院供应商编码、医院供应商名称、医院供应商信用代码、平台供应商信用代码、平台供应商名称、匹对状态、审批状态 |
| 输入 | {  "page":1,  "limit":10,  "hospitalName":"医院名称",  "supplyName":”医院供应商名称”,  "shipFlag":1  } |
| 业务代码 | 伪代码：Page.like(“hospitalName”, hospitalName).like(“supplyName”, supplyName).eq(“shipFlag”, shipFlag) |
| 查询语句 | 医院机构标识关联医院机构表、平台机构标识关联平台机构表 |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "shipId": 1,  "deptId": 1,  "horgId": 1,  "horgName": 医院名称,  "supplyNo": "药品名称",  "supplyName": "通用名称",  "supplyCode": “医院供应商信用代码”,  "porgId": "平台机构标识",  "porgName": "平台机构名称",  "shipFlag ": "20粒/盒",  "datasource": 2,  "delFlag": 1  },  {  "shipId": 2,  "deptId": 1,  "horgId": 1,  "horgName": 医院名称,  "supplyNo": "药品名称",  "supplyName": "通用名称",  "supplyCode": “医院供应商信用代码”,  "porgId": "平台机构标识",  "porgName": "平台机构名称",  "shipFlag ": "20粒/盒",  "datasource": 2,  "delFlag": 1  },  {  "shipId": 3,  "deptId": 1,  "horgId": 1,  "horgName": 医院名称,  "supplyNo": "药品名称",  "supplyName": "通用名称",  "supplyCode": “医院供应商信用代码”,  "porgId": "平台机构标识",  "porgName": "平台机构名称",  "shipFlag ": "20粒/盒",  "datasource": 2,  "delFlag": 1  }  ]  }  } |

### 5.2.2.2 新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 输入 | {  "horgId":1,  "supplyNo":”医院供应商编码”,  "supplyName":"医院供应商名称",  "supplyCode":”医院供应商信用代码”,  "porgId":1  } |
| 业务代码 | 伪代码：Page.like(“hospitalName”, hospitalName).like(“supplyName”, supplyName).eq(“shipFlag”, shipFlag) |
| 代码逻辑 | 1. 新增时。若根据医院供应商信用代码和平台供应商信用代码查询医院供应商匹配表可以查出记录，则不能新增。 2. 数据来源手工新增默认为2 3. 删除标志默认为0 4. 匹配标志默认为1（前台必须要求两个信用代码必填） 5. 新增记录默认修改人与创建人信息一致 |
| 输出 | {  "msg": "success",  "code": 0  } |

### 平台供应商分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 查询条件 | 平台供应商名称，平台供应商信用代码 |
| 输入 | {  "page":1,  "limit":10,  "porgName":"平台供应商名称",  "porgCode":”平台供应商信用代码,  } |
| 业务代码 | 伪代码：Page.like(“porgName”, porgName).eq(“porgCode”, porgCode)  只查询平台供应商表未匹配医院供应商的记录 |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [{  “porgId”:1  “porgName”:”平台供应商名称”,  “porgCode”:”平台供应商信用代码”  },  {  “porgId”:2  “porgName”:”平台供应商名称”,  “porgCode”:”平台供应商信用代码”  },  {  “porgId”:3  “porgName”:”平台供应商名称”,  “porgCode”:”平台供应商信用代码”  }]  }  } |

### 5.2.2.4 匹对查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 列表显示 | 后台返回字段：关系标识、医院供应商信用代码、医院供应商名称、平台供应商信用代码、平台供应商名称 |
| 输入 | {  "shipId":1,  "supplyName":”医院供应商名称”  } |
| 业务代码 | 配置匹配规则、注入MatchingRulesExecutor类调用matching方法拿到匹配结果。 |
| 输出 | {  "msg": "success",  "code": 0,  "data": {  "shipId": 1,  "supplyName": "医院供应商名称",  "supplyCode": “医院供应商信用代码”,  “list”:[{  "porgId": 1,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码"  “similarity”:1.00  },  {  "porgId":2,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码",  “similarity”:0.99  },  {  "porgId": 3,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码",  “similarity”:0.98  },  {  "porgId": 4,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码",  “similarity”:0.95  },  {  "porgId": 5,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码",  “similarity”:0.90  }]  }  } |

### 5.2.2.5 编辑保存

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 输入 | {  "shipId":1,  "porgId":2  } |
| 输出 | {  "msg": "success",  "code": 0  } |

### 5.2.2.6 删除

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 输入 | [1,2,3……] |
| 业务代码 | deleteBatchIds、修改删除标志del\_flag为1 |
| 输出 | {  "msg": "success",  "code": 0  } |

### 5.2.3商品匹配管理

### 5.2.3.1 分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 查询条件 | 医院名称/供应商名称、医院商品名称/供应商商品名称、生产厂商、匹配状态、审批状态 |
| 列表显示 | 1. 后台返回字段：关系标识、医院名称/供应商名称、医院商品名称/供应商商品名称、医院商品规格/供应商商品规格、生产厂商、批准文号/注册证号、平台商品名称、平台商品规格、匹对状态、审批状态 |
| 输入 | 医院商品匹配：  {  "page":1,  "limit":10,  "hospitalName":"医院名称",  "supplyName":”医院供应商名称”,  "shipFlag":1  }  供应商商品匹配：  {  "page":1,  "limit":10,  "hospitalName":"医院名称",  "supplyName":”医院供应商名称”,  "shipFlag":1  } |
| 业务代码 | 伪代码：Page.like(“hospitalName”, hospitalName).like(“supplyName”, supplyName).eq(“shipFlag”, shipFlag) |
| 查询语句 | 医院机构标识关联医院机构表、平台机构标识关联平台机构表 |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "shipId": 1,  "deptId": 1,  "horgId": 1,  "horgName": 医院名称,  "supplyNo": "药品名称",  "supplyName": "通用名称",  "supplyCode": “医院供应商信用代码”,  "porgId": "平台机构标识",  "porgName": "平台机构名称",  "shipFlag ": "20粒/盒",  "datasource": 2,  "delFlag": 1  },  {  "shipId": 2,  "deptId": 1,  "horgId": 1,  "horgName": 医院名称,  "supplyNo": "药品名称",  "supplyName": "通用名称",  "supplyCode": “医院供应商信用代码”,  "porgId": "平台机构标识",  "porgName": "平台机构名称",  "shipFlag ": "20粒/盒",  "datasource": 2,  "delFlag": 1  },  {  "shipId": 3,  "deptId": 1,  "horgId": 1,  "horgName": 医院名称,  "supplyNo": "药品名称",  "supplyName": "通用名称",  "supplyCode": “医院供应商信用代码”,  "porgId": "平台机构标识",  "porgName": "平台机构名称",  "shipFlag ": "20粒/盒",  "datasource": 2,  "delFlag": 1  }  ]  }  } |

### 5.2.2.2 新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 输入 | {  "horgId":1,  "supplyNo":”医院供应商编码”,  "supplyName":"医院供应商名称",  "supplyCode":”医院供应商信用代码”,  "porgId":1  } |
| 业务代码 | 伪代码：Page.like(“hospitalName”, hospitalName).like(“supplyName”, supplyName).eq(“shipFlag”, shipFlag) |
| 代码逻辑 | 1. 新增时。若根据医院供应商信用代码和平台供应商信用代码查询医院供应商匹配表可以查出记录，则不能新增。 2. 数据来源手工新增默认为2 3. 删除标志默认为0 4. 匹配标志默认为1（前台必须要求两个信用代码必填） 5. 新增记录默认修改人与创建人信息一致 |
| 输出 | {  "msg": "success",  "code": 0  } |

### 平台供应商分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 查询条件 | 平台供应商名称，平台供应商信用代码 |
| 输入 | {  "page":1,  "limit":10,  "porgName":"平台供应商名称",  "porgCode":”平台供应商信用代码,  } |
| 业务代码 | 伪代码：Page.like(“porgName”, porgName).eq(“porgCode”, porgCode)  只查询平台供应商表未匹配医院供应商的记录 |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [{  “porgId”:1  “porgName”:”平台供应商名称”,  “porgCode”:”平台供应商信用代码”  },  {  “porgId”:2  “porgName”:”平台供应商名称”,  “porgCode”:”平台供应商信用代码”  },  {  “porgId”:3  “porgName”:”平台供应商名称”,  “porgCode”:”平台供应商信用代码”  }]  }  } |

### 5.2.2.4 匹对查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 列表显示 | 后台返回字段：关系标识、医院供应商信用代码、医院供应商名称、平台供应商信用代码、平台供应商名称 |
| 输入 | {  "shipId":1,  "supplyName":”医院供应商名称”  } |
| 业务代码 | 配置匹配规则、注入MatchingRulesExecutor类调用matching方法拿到匹配结果。 |
| 输出 | {  "msg": "success",  "code": 0,  "data": {  "shipId": 1,  "supplyName": "医院供应商名称",  "supplyCode": “医院供应商信用代码”,  “list”:[{  "porgId": 1,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码"  “similarity”:1.00  },  {  "porgId":2,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码",  “similarity”:0.99  },  {  "porgId": 3,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码",  “similarity”:0.98  },  {  "porgId": 4,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码",  “similarity”:0.95  },  {  "porgId": 5,  "porgName": "平台机构名称",  "porgCode": "平台机构信用代码",  “similarity”:0.90  }]  }  } |

### 5.2.2.5 编辑保存

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 输入 | {  "shipId":1,  "porgId":2  } |
| 输出 | {  "msg": "success",  "code": 0  } |

### 5.2.2.6 删除

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 输入 | [1,2,3……] |
| 业务代码 | deleteBatchIds、修改删除标志del\_flag为1 |
| 输出 | {  "msg": "success",  "code": 0  } |

## 5.3商品管理

### 5.3.1平台药品信息

#### 5.3.1.1用例图



#### 5.3.1.2功能设计说明

##### 5.3.1.2.1分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 数据权限 | 根据当前用户角色分配的数据权限获取数据 |
| 查询条件 | 药品名称、商品属性、商品分类、状态、生产厂家、审批状态 |
| 列表显示 | 1、显示字段：商品统一编码、药品名称、商品属性、商品分类、商品规格、生产厂家、批准文号/注册证号、状态、审批状态  2、商品统一编码（系统自动按照统一编码规则生成）  3、排序：先按修改时间，再按创建时间倒序 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "page":1,  "limit":10,  "drugsName":"药品名称",  "goodsNature":0,  "typeId":1,  "status":1,  "factoryId":1  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  int currPage = Integer.parseInt(params.get("page").toString());  int pageSize = Integer.parseInt(params.get("limit").toString());  Page<GoodsPlatformDrugsEntityVo> page = new Page<GoodsPlatformDrugsEntityVo>(currPage, pageSize);  //多表关联查询  List<GoodsPlatformDrugsEntityVo> list = this.baseMapper.listForPage(page,params.get(Constant.SQL\_FILTER), params.get("drugsName"), params.get("goodsNature"), ......);  page.setRecords(list);  return new PageUtils(page);  } |
| 查询语句 | 平台药品表关联药品分类表、厂商信息表、平台药品规格表、平台药品批准文号表进行查询 |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": 1,  "goodsUnicode": "pt201905060001",  "drugsName": "药品名称",  "commonName": "通用名称",  "goodsNature": 0,  "typeName": "分类名称",  "specs": "20粒/盒",  "specsList": [  {  "id": 1,  "specs": "8袋/盒",  "status": 1,  ......  },  {  "id": 2,  "specs": "10袋/盒",  "status": 0,  ......  },  {  "id": 3,  "specs": "10袋/盒",  "status": 1,  ......  }  ],  "factoryName": "广州白云山",  "approvalsList": [  {  "id": 1,  "approvals": "国药准字H10970001",  "status": 1,  ......  },  {  "id": 2,  "approvals": "国药准字H10970002",  "status": 0,  ......  },  {  "id": 3,  "specs": "国药准字H10970003",  "status": 1,  ......  }  ],  "status": 1,  ......  },  {  "id": 2,  "goodsUnicode": "pt201905060002",  "drugsName": "药品名称",  "commonName": "通用名称",  "goodsNature": 0,  "typeName": "分类名称",  "specsList": [  {  "id": 1,  "specs": "8袋/盒",  "status": 1,  ......  },  {  "id": 2,  "specs": "10袋/盒",  "status": 0,  ......  },  {  "id": 3,  "specs": "10袋/盒",  "status": 1,  ......  }  ],  "factoryName": "广东罗浮山",  "approvalsList": [  {  "id": 1,  "approvals": "国药准字H10970001",  "status": 1,  ......  },  {  "id": 2,  "approvals": "国药准字H10970002",  "status": 0,  ......  },  {  "id": 3,  "specs": "国药准字H10970003",  "status": 1,  ......  }  ],  "status": 1,  ......  }  ]  }  } |

##### 5.3.1.2.2新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、商品统一编码（系统自动按照统一编码规则生成，参考统一编码规范V4.0）  2、药品名称，唯一性约束  3、生产厂商存在（厂商id和厂商名字必填），否则只传厂商名称  4、商品下的规格和批准文号，唯一性约束 |
| 必填字段 | 药品名称、商品属性、药品分类、药品规格、生产厂商、批准文号、状态、商品单位 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "drugsName":"感冒灵颗粒",  "commonName": "感冒灵",  "goodsNature":0,  "typeId":1,  "factoryId":1,  "factoryName": "广州白云山",  "status":1,  "goodsUnit":"1",  "specs":"8袋/盒",  "guid":"guid00000001",  "approvals": "国药准字H10970001,国药准字H10970002,国药准字H10970003",  .....  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void save(GoodsPlatformDrugsEntityVo goodsPlatformDrugsEntityVo) {  List<GoodsPlatformDrugsEntityVo> drugsName = baseMapper.selectByDrugsName(goodsPlatformDrugsEntityVo.getDrugsName());  if(!StringUtil.isEmpty(drugsName)){  throw new HdiException("药品名称已存在");  }  //判断药品规格是否存在（药品下规格唯一）  ....  //判断批准文号是否已经存在（药品下规格唯一）  .....  //生产厂商id为空则生产厂商信息  if(StringUtil.isEmpty(goodsPlatformDrugsEntityVo.getFactoryId())){  //保存生产厂商信息，状态为草稿  .......  }  //生产商品统一编码  .......  //上传商品图片到fastDFS  ......  //保存药品信息  goodsPlatformDrugsEntityVo.setCreateTime(new Date());  this.insert(goodsPlatformDrugsEntityVo);  //保存商品规格  ......  //保存批准文号信息  ......  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.3.1.2.3编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、药品名称，唯一性约束  2、生产厂商存在（厂商id和厂商名字必填），否则只传厂商名称  3、批准文号只能修改、禁用、不能删除，商品下的批准文号唯一性约束  4、编辑是没有商品规格和全球唯一码字段 |
| 必填字段 | 药品名称、商品属性、药品分类、生产厂商、批准文号、状态、商品单位 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "id":3,  "drugsName":"感冒灵颗粒",  "commonName": "感冒灵",  "goodsNature":0,  "typeId":1,  "factoryId":1,  "factoryName": "广州白云山",  "approvalsList": [  {  "id": 1,  "approvals":"国药准字H10970001" ,  "status": 1,  ......  },  {  "id": 2,  "approvals":"国药准字H10970002" ,  "status": 0,  ......  },  {  "id": null,  "specs":"国药准字H10970003" ,  "status": null,  ......  }  ],  "status":1,  "guid":"guid00000001",  "goodsUnit":"1",  .....  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void update(GoodsPlatformDrugsEntityVo goodsPlatformDrugsEntityVo) {  //判断药品名称是否存在  if(存在){  throw new HdiException("药品名称已存在");  }  //判断批准文号是否已经存在（药品下规格唯一）  .....  //生产厂商id为空则生产厂商信息  if(StringUtil.isEmpty(goodsPlatformDrugsEntityVo.getFactoryId())){  //保存生产厂商信息，状态为草稿  .......  }  //图片路径为空则上传商品图片到fastDFS,否则不上传  ......  //更新药品信息  goodsPlatformDrugsEntityVo.setEditTime(new Date());  this.updateById(goodsPlatformDrugsEntityVo);  //更新批准文号或保存批准文号信息  ......  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.3.1.2.4查询详情

|  |  |
| --- | --- |
| 事项 | 内容 |
| 详情显示 | 商品统一编码、药品名称、药品通用名称、商品属性、商品分类名称、商品规格集合、生产厂商名称、批准文号集合、状态、商品单位、存储方式、商品图片URL |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | http://localhost:2320/hdi-server-admin/drugs/goodsplatformdrugs/info/1 |
| 业务代码 | @RequestMapping("/info/{id}")  public Hdi info(@PathVariable("id") Long id){  GoodsPlatformDrugsEntityVo goodsPlatformDrugsEntityVo = goodsPlatformDrugsService.selectPlatformDrugsById(id);  return Hdi.ok().put("goodsPlatformDrugs", goodsPlatformDrugsEntityVo );  } |
| 查询语句 | 平台药品表关联药品分类表、厂商信息表、平台药品规格表、平台药品批准文号表进行查询 |
| 输出 | {  "msg": "success",  "code": 0,  "goodsPlatformDrugs": {  "id": 3,  "drugsName": "感冒灵颗粒",  "commonName": "感冒灵",  "goodsNature": 0,  "typeId": 1,  "specsList": [  {  "id": 1,  "specs": "8袋/盒",  "status": 1,  ......  },  {  "id": 2,  "specs": "10袋/盒",  "status": 0,  ......  },  {  "id": 3,  "specs": "10袋/盒",  "status": 1,  ......  }  ],  "factoryId": 1,  "factoryName": "广州白云山",  "approvalsList": [  {  "id": 1,  "approvals": "国药准字H10970001",  "status": 1,  ......  },  {  "id": 2,  "approvals": "国药准字H10970002",  "status": 0,  ......  },  {  "id": 3,  "specs": "国药准字H10970003",  "status": 1,  ......  }  ],  "status": 1,  "guid": "guid00000001",  "goodsUnit": "1",  .....  }  } |

##### 5.3.1.2.5删除

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.drugs |
| 功能说明 | 做软删除，更新状态操作 |
| 输入 | [1,2,3] |
| 业务代码 | @Override  public void delete(Long[] ids) {  for(Long id : ids) {  GoodsPlatformDrugsEntity goodsPlatformDrugsEntity = new GoodsPlatformDrugsEntity ();  goodsPlatformDrugsEntity .setId(id);  //设置信息为删除状态  goodsPlatformDrugsEntity.setStatus(StatusEnum.DELETE.getKey());  this.updateById(goodsPlatformDrugsEntity );  }  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.3.1.2.6更多规格

###### 5.3.1.2.6.1分页列表

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 查询条件 | 平台药品规格（模糊匹配）、状态 |
| 列表显示 | 平台药品规格、全球唯一编码、状态 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "page":"1",  "limit":"10",  "specs":"8袋/盒",  "status":"1"  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  Page<GoodsPlatformDrugsSpecsEntity> page = this.selectPage(  new Query<GoodsPlatformDrugsSpecsEntity>(params).getPage(),  new EntityWrapper<GoodsPlatformDrugsSpecsEntity>()  .addFilterIfNeed(params.get(Constant.SQL\_FILTER) != null, (String)params.get(Constant.SQL\_FILTER))  .like(StringUtils.isNotBlank(params.get("specs").toString()), "specs", params.get("specs").toString())  .eq(StringUtils.isNotBlank(params.get("status").toString()), "status", params.get("status").toString())  );  return new PageUtils(page);  } |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": 1,  "drugsId": 1,  "specsCode": "gg00001",  "specs": "8袋/盒",  "guid": "guid00001",  "status": "1",  ......  },  {  "id": 2,  "drugsId": 2,  "specsCode": "gg00002",  "specs": "10袋/盒",  "guid": "guid00002",  "status": "1",  ......  }......  ]  }  } |

###### 5.3.1.2.6.1编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1. 新增记录可以删除，修改记录不允许删除 2. 支持批量保存新增记录和更新记录 |
| 必填字段 | 平台药品ID、平台药品规格、全球唯一码、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 业务代码 | [  {  "id": 1,  "drugsId": 1,  "specsCode": "gg00001",  "specs": "8袋/盒",  "guid": "guid00001",  "status": "1",  ......  },  {  "id": null,  "drugsId": null,  "specsCode": "",  "specs": "10袋/盒",  "guid": "guid00002",  "status": "1",  ......  }......  ] |
| 输出 | @Override  @Transactional(rollbackFor = Exception.class)  public void update(List<GoodsPlatformDrugsSpecsEntity> goodsPlatformDrugsSpecsList) {  for(GoodsPlatformDrugsSpecsEntity goodsPlatformDrugsSpecsEntity ： goodsPlatformDrugsSpecsList){  //判断规格是否存在(商品下规格唯一)  if(存在){  throw new HdiException("药品规格已存在");  }  }  for(GoodsPlatformDrugsSpecsEntity goodsPlatformDrugsSpecsEntity ： goodsPlatformDrugsSpecsList){  if(StringUtil.isEmpty(goodsPlatformDrugsSpecsEntity.getId())){  this.insert(goodsPlatformDrugsSpecsEntity);  }else{  this.updateById(goodsPlatformDrugsSpecsEntity);  }  }  } |
| 业务代码 | {  "msg": "success",  "code": 0  } |

### 5.3.2供应商药品信息

#### 5.3.2.1用例图



#### 5.3.2.2功能设计说明

##### 5.3.2.2.1分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 查询条件 | 药品名称、商品属性、商品分类、状态、生产厂家、供应商名称 |
| 列表显示 | 1、显示字段：供应商名称、药品名称、商品属性、商品分类、商品规格、生产厂家、批准文号/注册证号、状态、审批状态  2、商品统一编码（系统自动按照统一编码规则生成）  3、排序：先按修改时间，再按创建时间倒序 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "page":1,  "limit":10,  "drugsName":"药品名称",  "goodsNature":0,  "typeId":1,  "status":1,  "factoryId":1  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  int currPage = Integer.parseInt(params.get("page").toString());  int pageSize = Integer.parseInt(params.get("limit").toString());  Page<GoodsSupplierDrugsEntityVo> page = new Page<GoodsSupplierDrugsEntityVo>(currPage, pageSize);  //多表关联查询  List<GoodsSupplierDrugsEntityVo> list = this.baseMapper.listForPage(page,params.get(Constant.SQL\_FILTER), params.get("drugsName"), params.get("goodsNature"), ......);  page.setRecords(list);  return new PageUtils(page);  } |
| 查询语句 | 供应商药品表关联药品分类表、厂商信息表、供应商药品规格表、供应商药品批准文号表进行查询 |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": 1,  "drugsName": "药品名称",  "commonName": "通用名称",  "goodsNature": 0,  "typeName": "分类名称",  "specs": "20粒/盒",  "specsList": [  {  "id": 1,  "specs": "8袋/盒",  "status": 1,  ......  },  {  "id": 2,  "specs": "10袋/盒",  "status": 0,  ......  },  {  "id": 3,  "specs": "10袋/盒",  "status": 1,  ......  }  ],  "factoryName": "广州白云山",  "approvalsList": [  {  "id": 1,  "approvals": "国药准字H10970001",  "status": 1,  ......  },  {  "id": 2,  "approvals": "国药准字H10970002",  "status": 0,  ......  },  {  "id": 3,  "specs": "国药准字H10970003",  "status": 1,  ......  }  ],  "status": 1,  ......  },  {  "id": 2,  "drugsName": "药品名称",  "commonName": "通用名称",  "goodsNature": 0,  "typeName": "分类名称",  "specsList": [  {  "id": 1,  "specs": "8袋/盒",  "status": 1,  ......  },  {  "id": 2,  "specs": "10袋/盒",  "status": 0,  ......  },  {  "id": 3,  "specs": "10袋/盒",  "status": 1,  ......  }  ],  "factoryName": "广东罗浮山",  "approvalsList": [  {  "id": 1,  "approvals": "国药准字H10970001",  "status": 1,  ......  },  {  "id": 2,  "approvals": "国药准字H10970002",  "status": 0,  ......  },  {  "id": 3,  "specs": "国药准字H10970003",  "status": 1,  ......  }  ],  "status": 1,  ......  }  ]  }  } |

##### 5.3.2.2.2新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、当前登陆用户为供应商，则供应商名称默认不能选择；如果是平台管理员，则供应商名称可以选择  2、药品名称，在供应商范围内唯一性约束  3、生产厂商存在（厂商id和厂商名字必填），否则只传厂商名称  4、商品下的规格和批准文号，唯一性约束 |
| 必填字段 | 供应商名称、药品名称、商品属性、药品分类、药品规格、生产厂商、批准文号、状态 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "supplierId":1,  "drugsName":"感冒灵颗粒",  "commonName": "感冒灵",  "goodsNature":0,  "typeId":1,  "specs":"8袋/盒",  "factoryId":1,  "factoryName": "广州白云山",  "approvals": "国药准字H10970001,国药准字H10970002,国药准字H10970003",  "status":1,  "guid":"guid00000001",  "goodsUnit":"1",  .....  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void save(GoodsSupplierDrugsEntityVo goodsSupplierDrugsEntityVo) {  List<GoodsSupplierDrugsEntityVo> drugsName = baseMapper.selectBySupplierIdAndDrugsName(goodsSupplierDrugsEntityVo.getSupplierId(), goodsSupplierDrugsEntityVo.getDrugsName());  if(!StringUtil.isEmpty(drugsName)){  throw new HdiException("药品名称已存在");  }  //判断药品规格是否存在（药品下规格唯一）  ....  //判断批准文号是否已经存在（药品下规格唯一）  .....  //生产厂商id为空则生产厂商信息  if(StringUtil.isEmpty(goodsSupplierDrugsEntityVo.getFactoryId())){  //保存生产厂商信息，状态为草稿  .......  }  //生产商品统一编码  .......  //上传商品图片到fastDFS  ......  //保存药品信息  goodsSupplierDrugsEntityVo.setCreateTime(new Date());  this.insert(goodsPlatformDrugsEntityVo);  //保存商品规格  ......  //保存批准文号信息  ......  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.3.2.2.3编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、药品名称，唯一性约束  2、生产厂商存在（厂商id和厂商名字必填），否则只传厂商名称  3、批准文号只能修改、禁用、不能删除，商品下的批准文号唯一性约束  4、编辑是没有商品规格和全球唯一码字段 |
| 必填字段 | 药品名称、商品属性、药品分类、药品规格、生产厂商、批准文号、状态、商品单位 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "id":3,  "supplierId":1,  "drugsName":"感冒灵颗粒",  "commonName": "感冒灵",  "goodsNature":0,  "typeId":1,  "factoryId":1,  "factoryName": "广州白云山",  "approvalsList": [  {  "id": 1,  "approvals":"国药准字H10970001" ,  "status": 1,  ......  },  {  "id": 2,  "approvals":"国药准字H10970002" ,  "status": 0,  ......  },  {  "id": null,  "specs":"国药准字H10970003" ,  "status": null,  ......  }  ],  "status":1,  "guid":"guid00000001",  "goodsUnit":"1",  .....  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void update(GoodsSupplierDrugsEntityVo goodsSupplierDrugsEntityVo ) {  //判断药品名称是否存在  if(存在){  throw new HdiException("药品名称已存在");  }  //判断批准文号是否已经存在（药品下规格唯一）  .....  //生产厂商id为空则生产厂商信息  if(StringUtil.isEmpty(goodsSupplierDrugsEntityVo.getFactoryId())){  //保存生产厂商信息，状态为草稿  .......  }  //图片路径为空则上传商品图片到fastDFS,否则不上传  ......  //更新药品信息  goodsSupplierDrugsEntityVo.setEditTime(new Date());  this.updateById(goodsSupplierDrugsEntityVo);  //更新药品规格或保存药品规格  .....  //更新批准文号或保存批准文号信息  ......  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.3.1.2.4查询详情

|  |  |
| --- | --- |
| 事项 | 内容 |
| 详情显示 | 药品名称、药品通用名称、商品属性、商品分类名称、商品规格集合、生产厂商名称、批准文号集合、状态、商品单位、存储方式、商品图片URL |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | http://localhost:2320/hdi-server-admin/drugs/goodssupplierdrugs/info/1 |
| 业务代码 | @RequestMapping("/info/{id}")  public Hdi info(@PathVariable("id") Long id){  GoodsSupplierDrugsEntityVo goodsSupplierDrugsEntityVo= goodsSupplierDrugsService.selectSupplierDrugsById(id);  return Hdi.ok().put("goodsSupplierDrugs", goodsSupplierDrugsEntityVo);  } |
| 查询语句 | 供应商药品表关联药品分类表、厂商信息表、供应商药品规格表、平台药品批准文号表进行查询 |
| 输出 | {  "msg": "success",  "code": 0,  "goodsPlatformDrugs": {  "id": 3,  "supplierId":1,  "drugsName": "感冒灵颗粒",  "commonName": "感冒灵",  "goodsNature": 0,  "typeId": 1,  "specsList": [  {  "id": 1,  "specs": "8袋/盒",  "status": 1,  ......  },  {  "id": 2,  "specs": "10袋/盒",  "status": 0,  ......  },  {  "id": 3,  "specs": "10袋/盒",  "status": 1,  ......  }  ],  "factoryId": 1,  "factoryName": "广州白云山",  "approvalsList": [  {  "id": 1,  "approvals": "国药准字H10970001",  "status": 1,  ......  },  {  "id": 2,  "approvals": "国药准字H10970002",  "status": 0,  ......  },  {  "id": 3,  "specs": "国药准字H10970003",  "status": 1,  ......  }  ],  "status": 1,  "guid": "guid00000001",  "goodsUnit": "1",  .....  }  } |

##### 5.3.2.2.5删除

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.drugs |
| 功能说明 | 做软删除，更新状态操作 |
| 输入 | [1,2,3] |
| 业务代码 | @Override  public void delete(Long[] ids) {  for(Long id : ids) {  GoodsSupplierDrugsEntityVo goodsSupplierDrugsEntity = new GoodsSupplierDrugsEntity ();  goodsSupplierDrugsEntity.setId(id);  //设置信息为删除状态  goodsSupplierDrugsEntity.setStatus(StatusEnum.DELETE.getKey());  this.updateById(goodsSupplierDrugsEntity);  }  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.3.2.2.6更多规格

###### 5.3.2.2.6.1分页列表

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 查询条件 | 供应商药品规格（模糊匹配）、状态 |
| 列表显示 | 供应商药品规格、全球唯一编码、状态 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "page":"1",  "limit":"10",  "specs":"8袋/盒",  "status":"1"  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  Page<GoodsSupplierDrugsSpecsEntity> page = this.selectPage(  new Query<GoodsSupplierDrugsSpecsEntity>(params).getPage(),  new EntityWrapper<GoodsSupplierDrugsSpecsEntity>()  .addFilterIfNeed(params.get(Constant.SQL\_FILTER) != null, (String)params.get(Constant.SQL\_FILTER))  .like(StringUtils.isNotBlank(params.get("specs").toString()), "specs", params.get("specs").toString())  .eq(StringUtils.isNotBlank(params.get("status").toString()), "status", params.get("status").toString())  );  return new PageUtils(page);  } |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": 1,  "drugsId": 1,  "specsCode": "gg00001",  "specs": "8袋/盒",  "guid": "guid00001",  "status": "1",  ......  },  {  "id": 2,  "drugsId": 2,  "specsCode": "gg00002",  "specs": "10袋/盒",  "guid": "guid00002",  "status": "1",  ......  }......  ]  }  } |

###### 5.3.2.2.6.1编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1. 新增记录可以删除，修改记录不允许删除 2. 支持批量保存新增记录和更新记录 |
| 必填字段 | 供应商药品ID、供应商药品规格、全球唯一码、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 业务代码 | [  {  "id": 1,  "drugsId": 1,  "specsCode": "gg00001",  "specs": "8袋/盒",  "guid": "guid00001",  "status": "1",  ......  },  {  "id": null,  "drugsId": null,  "specsCode": "",  "specs": "10袋/盒",  "guid": "guid00002",  "status": "1",  ......  }......  ] |
| 输出 | @Override  @Transactional(rollbackFor = Exception.class)  public void update(List<GoodsSupplierDrugsSpecsEntity> goodsSupplierDrugsSpecsList) {  for(GoodsSupplierDrugsSpecsEntity goodsSupplierDrugsSpecsEntity ： goodsSupplierDrugsSpecsList){  //判断规格是否存在(商品下规格唯一)  if(存在){  throw new HdiException("药品规格已存在");  }  }  for(GoodsSupplierDrugsSpecsEntity goodsSupplierDrugsSpecsEntity ： goodsSupplierDrugsSpecsList){  if(StringUtil.isEmpty(goodsSupplierDrugsSpecsEntity.getId())){  this.insert(goodsSupplierDrugsSpecsEntity);  }else{  this.updateById(goodsSupplierDrugsSpecsEntity);  }  }  } |
| 业务代码 | {  "msg": "success",  "code": 0  } |

### 5.3.3医院药品信息

#### 5.3.3.1用例图



#### 5.3.3.2功能设计说明

##### 5.3.3.2.1分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 查询条件 | 药品名称、商品属性、商品分类、状态、生产厂家、供应商名称 |
| 列表显示 | 1、显示字段：医院名称、药品名称、商品属性、商品分类、商品规格、生产厂家、批准文号/注册证号、状态、审批状态  2、商品统一编码（系统自动按照统一编码规则生成）  3、排序：先按修改时间，再按创建时间倒序 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "page":1,  "limit":10,  "drugsName":"药品名称",  "goodsNature":0,  "typeId":1,  "status":1,  "factoryId":1  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  int currPage = Integer.parseInt(params.get("page").toString());  int pageSize = Integer.parseInt(params.get("limit").toString());  Page<GoodsHospitalDrugsEntityVo> page = new Page<GoodsHospitalDrugsEntityVo>(currPage, pageSize);  //多表关联查询  List<GoodsHospitalDrugsEntityVo> list = this.baseMapper.listForPage(page,params.get(Constant.SQL\_FILTER), params.get("drugsName"), params.get("goodsNature"), ......);  page.setRecords(list);  return new PageUtils(page);  } |
| 查询语句 | 医院药品表关联药品分类表、厂商信息表、医院药品规格表、医院药品批准文号表进行查询 |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": 1,  "drugsName": "药品名称",  "commonName": "通用名称",  "goodsNature": 0,  "typeName": "分类名称",  "specs": "20粒/盒",  "specsList": [  {  "id": 1,  "specs": "8袋/盒",  "status": 1,  ......  },  {  "id": 2,  "specs": "10袋/盒",  "status": 0,  ......  },  {  "id": 3,  "specs": "10袋/盒",  "status": 1,  ......  }  ],  "factoryName": "广州白云山",  "approvalsList": [  {  "id": 1,  "approvals": "国药准字H10970001",  "status": 1,  ......  },  {  "id": 2,  "approvals": "国药准字H10970002",  "status": 0,  ......  },  {  "id": 3,  "specs": "国药准字H10970003",  "status": 1,  ......  }  ],  "status": 1,  ......  },  {  "id": 2,  "drugsName": "药品名称",  "commonName": "通用名称",  "goodsNature": 0,  "typeName": "分类名称",  "specsList": [  {  "id": 1,  "specs": "8袋/盒",  "status": 1,  ......  },  {  "id": 2,  "specs": "10袋/盒",  "status": 0,  ......  },  {  "id": 3,  "specs": "10袋/盒",  "status": 1,  ......  }  ],  "factoryName": "广东罗浮山",  "approvalsList": [  {  "id": 1,  "approvals": "国药准字H10970001",  "status": 1,  ......  },  {  "id": 2,  "approvals": "国药准字H10970002",  "status": 0,  ......  },  {  "id": 3,  "specs": "国药准字H10970003",  "status": 1,  ......  }  ],  "status": 1,  ......  }  ]  }  } |

##### 5.3.3.2.2新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、药品名称，在供应商范围内唯一性约束  2、生产厂商存在（厂商id和厂商名字必填），否则只传厂商名称  3、商品下的规格和批准文号，唯一性约束 |
| 必填字段 | 医院名称、药品名称、商品属性、药品分类、药品规格、生产厂商、批准文号、状态 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "hospitalId":1,  "drugsName":"感冒灵颗粒",  "commonName": "感冒灵",  "goodsNature":0,  "typeId":1,  "specs":"8袋/盒",  "factoryId":1,  "factoryName": "广州白云山",  "approvals": "国药准字H10970001,国药准字H10970002,国药准字H10970003",  "status":1,  "guid":"guid00000001",  "goodsUnit":"1",  .....  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void save(GoodsHospitalDrugsEntityVo goodsHospitalDrugsEntityVo) {  List<GoodsHospitalDrugsEntityVo> drugsName = baseMapper.selectByHospitalIdAndDrugsName(goodsHospitalDrugsEntityVo.getHospitalId(), goodsHospitalDrugsEntityVo.getDrugsName());  if(!StringUtil.isEmpty(drugsName)){  throw new HdiException("药品名称已存在");  }  //判断药品规格是否存在（药品下规格唯一）  ....  //判断批准文号是否已经存在（药品下规格唯一）  .....  //生产厂商id为空则生产厂商信息  if(StringUtil.isEmpty(goodsHospitalDrugsEntityVo.getFactoryId())){  //保存生产厂商信息，状态为草稿  .......  }  //生产商品统一编码  .......  //上传商品图片到fastDFS  ......  //保存药品信息  goodsHospitalDrugsEntityVo.setCreateTime(new Date());  this.insert(goodsHospitalDrugsEntityVo);  //保存商品规格  ......  //保存批准文号信息  ......  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.3.3.2.3编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1、药品名称，唯一性约束  2、生产厂商存在（厂商id和厂商名字必填），否则只传厂商名称  3、批准文号只能修改、禁用、不能删除，商品下的批准文号唯一性约束  4、编辑是没有商品规格和全球唯一码字段 |
| 必填字段 | 药品名称、商品属性、药品分类、药品规格、生产厂商、批准文号、状态、商品单位 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "id":3,  "hospitalId":1,  "drugsName":"感冒灵颗粒",  "commonName": "感冒灵",  "goodsNature":0,  "typeId":1,  "factoryId":1,  "factoryName": "广州白云山",  "approvalsList": [  {  "id": 1,  "approvals":"国药准字H10970001" ,  "status": 1,  ......  },  {  "id": 2,  "approvals":"国药准字H10970002" ,  "status": 0,  ......  },  {  "id": null,  "specs":"国药准字H10970003" ,  "status": null,  ......  }  ],  "status":1,  "guid":"guid00000001",  "goodsUnit":"1",  .....  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void update(GoodsHospitalDrugsEntityVo goodsHospitalDrugsEntityVo) {  //判断药品名称是否存在  if(存在){  throw new HdiException("药品名称已存在");  }  //判断批准文号是否已经存在（药品下规格唯一）  .....  //生产厂商id为空则生产厂商信息  if(StringUtil.isEmpty(goodsHospitalDrugsEntityVo.getFactoryId())){  //保存生产厂商信息，状态为草稿  .......  }  //图片路径为空则上传商品图片到fastDFS,否则不上传  ......  //更新药品信息  goodsHospitalDrugsEntityVo.setEditTime(new Date());  this.updateById(goodsHospitalDrugsEntityVo);  //更新药品规格或保存药品规格  .....  //更新批准文号或保存批准文号信息  ......  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.3.3.2.4查询详情

|  |  |
| --- | --- |
| 事项 | 内容 |
| 详情显示 | 药品名称、药品通用名称、商品属性、商品分类名称、商品规格集合、生产厂商名称、批准文号集合、状态、商品单位、存储方式、商品图片URL |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | http://localhost:2320/hdi-server-admin/drugs/goodshospitaldrugs/info/1 |
| 业务代码 | @RequestMapping("/info/{id}")  public Hdi info(@PathVariable("id") Long id){  GoodsHospitalDrugsEntityVo goodsHospitalDrugsEntityVo= goodsHospitalDrugsService.selectHospitalDrugsById(id);  return Hdi.ok().put("goodsHospitalDrugs", goodsHospitalDrugsEntityVo);  } |
| 查询语句 | 医院药品表关联药品分类表、厂商信息表、医院药品规格表、平台药品批准文号表进行查询 |
| 输出 | {  "msg": "success",  "code": 0,  "goodsPlatformDrugs": {  "id": 3,  "hospitalId":1,  "drugsName": "感冒灵颗粒",  "commonName": "感冒灵",  "goodsNature": 0,  "typeId": 1,  "specsList": [  {  "id": 1,  "specs": "8袋/盒",  "status": 1,  ......  },  {  "id": 2,  "specs": "10袋/盒",  "status": 0,  ......  },  {  "id": 3,  "specs": "10袋/盒",  "status": 1,  ......  }  ],  "factoryId": 1,  "factoryName": "广州白云山",  "approvalsList": [  {  "id": 1,  "approvals": "国药准字H10970001",  "status": 1,  ......  },  {  "id": 2,  "approvals": "国药准字H10970002",  "status": 0,  ......  },  {  "id": 3,  "specs": "国药准字H10970003",  "status": 1,  ......  }  ],  "status": 1,  "guid": "guid00000001",  "goodsUnit": "1",  .....  }  } |

##### 5.3.3.2.5删除

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.drugs |
| 功能说明 | 做软删除，更新状态操作 |
| 输入 | [1,2,3] |
| 业务代码 | @Override  public void delete(Long[] ids) {  for(Long id : ids) {  GoodsHospitalDrugsEntity goodsHospitalDrugsEntity= new GoodsHospitalDrugsEntity ();  goodsHospitalDrugsEntity.setId(id);  //设置信息为删除状态  goodsHospitalDrugsEntity.setStatus(StatusEnum.DELETE.getKey());  this.updateById(goodsHospitalDrugsEntity);  }  } |
| 输出 | {  "msg": "success",  "code": 0  } |

##### 5.3.3.2.6更多规格

###### 5.3.3.2.6.1分页列表

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 查询条件 | 医院药品规格（模糊匹配）、状态 |
| 列表显示 | 医院药品规格、全球唯一编码、状态 |
| 类包 | com.ebig.hdi.modules.drugs |
| 输入 | {  "page":"1",  "limit":"10",  "specs":"8袋/盒",  "status":"1"  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  Page<GoodsHospitalDrugsSpecsEntity> page = this.selectPage(  new Query<GoodsHospitalDrugsSpecsEntity>(params).getPage(),  new EntityWrapper<GoodsHospitalDrugsSpecsEntity>()  .addFilterIfNeed(params.get(Constant.SQL\_FILTER) != null, (String)params.get(Constant.SQL\_FILTER))  .like(StringUtils.isNotBlank(params.get("specs").toString()), "specs", params.get("specs").toString())  .eq(StringUtils.isNotBlank(params.get("status").toString()), "status", params.get("status").toString())  );  return new PageUtils(page);  } |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": 1,  "drugsId": 1,  "specsCode": "gg00001",  "specs": "8袋/盒",  "guid": "guid00001",  "status": "1",  ......  },  {  "id": 2,  "drugsId": 2,  "specsCode": "gg00002",  "specs": "10袋/盒",  "guid": "guid00002",  "status": "1",  ......  }......  ]  }  } |

###### 5.3.3.2.6.1编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 页面效果 |  |
| 约束条件 | 1. 新增记录可以删除，修改记录不允许删除 2. 支持批量保存新增记录和更新记录 |
| 必填字段 | 医院药品ID、供应商药品规格、全球唯一码、状态 |
| 类包 | com.ebig.hdi.modules.org |
| 业务代码 | [  {  "id": 1,  "drugsId": 1,  "specsCode": "gg00001",  "specs": "8袋/盒",  "guid": "guid00001",  "status": "1",  ......  },  {  "id": null,  "drugsId": null,  "specsCode": "",  "specs": "10袋/盒",  "guid": "guid00002",  "status": "1",  ......  }......  ] |
| 输出 | @Override  @Transactional(rollbackFor = Exception.class)  public void update(List<GoodsHospitalDrugsSpecsEntity> goodsHospitalDrugsSpecsList) {  for(GoodsHospitalDrugsSpecsEntity goodsHospitalDrugsSpecsEntity ： goodsHospitalDrugsSpecsList){  //判断规格是否存在(商品下规格唯一)  if(存在){  throw new HdiException("药品规格已存在");  }  }  for(GoodsHospitalDrugsSpecsEntity goodsHospitalDrugsSpecsEntity ： goodsHospitalDrugsSpecsList){  if(StringUtil.isEmpty(goodsHospitalDrugsSpecsEntity.getId())){  this.insert(goodsHospitalDrugsSpecsEntity);  }else{  this.updateById(goodsHospitalDrugsSpecsEntity);  }  }  } |
| 业务代码 | {  "msg": "success",  "code": 0  } |

### 5.3.4平台试剂信息

#### 5.3.4.1用例图



#### 5.3.4.2功能设计说明

##### 5.3.4.2.1分页查询

##### 5.3.4.2.2新增

##### 5.3.4.2.3编辑

##### 5.3.4.2.4查询详情

##### 5.3.4.2.5删除

### 5.3.5供应商试剂信息

#### 5.3.5.1用例图



#### 5.3.5.2功能设计说明

##### 5.3.5.2.1分页查询

##### 5.3.5.2.2新增

##### 5.3.5.2.3编辑

##### 5.3.5.2.4查询详情

##### 5.3.5.2.5删除

### 5.3.6医院试剂信息

#### 5.3.6.1用例图



#### 5.3.6.2功能设计说明

##### 5.3.6.2.1分页查询

##### 5.3.6.2.2新增

##### 5.3.6.2.3编辑

##### 5.3.6.2.4查询详情

##### 5.3.6.2.5删除

### 5.3.7平台耗材信息

#### 5.3.7.1用例图



#### 5.3.7.2功能设计说明

##### 5.3.7.2.1分页查询

##### 5.3.7.2.2新增

##### 5.3.7.2.3编辑

##### 5.3.7.2.4查询详情

##### 5.3.7.2.5删除

### 5.3.8供应商耗材信息

#### 5.3.8.1用例图



#### 5.3.8.2功能设计说明

##### 5.3.8.2.1分页查询

##### 5.3.8.2.2新增

##### 5.3.8.2.3编辑

##### 5.3.8.2.4查询详情

##### 5.3.8.2.5删除

### 5.3.9医院耗材信息

#### 5.3.9.1用例图



#### 5.3.9.2功能设计说明

##### 5.3.9.2.1分页查询

##### 5.3.9.2.2新增

##### 5.3.9.2.3编辑

##### 5.3.9.2.4查询详情

##### 5.3.9.2.5删除

## 5.4证照管理

### 5.4.1用例图

### 5.4.2功能设计说明

## 5.5采购管理

### 5.5.1用例图

### 5.5.2功能设计说明

## 5.6供货管理

### 5.6.1用例图

### 5.6.2功能设计说明

## 5.7退货管理

### 5.7.1用例图

### 5.7.1功能设计说明

## 5.8公共模块

### 5.8.1字典查询设计

#### 5.8.1.1根据类型查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 查询条件 | 字典类型 |
| 列表显示 | 字典类型、父字典码、字典码、字典值 |
| 类包 | com.ebig.hdi.modules.sys |
| 输入 | {  "type":"province,city,area"  } |
| 业务代码 | @Override  public List<SysDictEntity> selectDictByType(String type) {  return sysDictDao.selectDictByType(StringUtil.getInParam(type));  } |
| 输出 | {  "msg": "success",  "code": 0,  "dictList": [  {  "id": null,  "name": null,  "type": "province",  "parentCode": "",  "code": "11",  "value": "北京市",  "orderNum": null,  "remark": null,  "delFlag": null  },  {  "id": null,  "name": null,  "type": "province",  "parentCode": "",  "code": "12",  "value": "天津市",  "orderNum": null,  "remark": null,  "delFlag": null  }…  ]  } |

#### 5.8.1.2根据类型和父字典码查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 查询条件 | 字典类型、父字典码 |
| 列表显示 | 字典类型、父字典码、字典码、字典值 |
| 类包 | com.ebig.hdi.modules.sys |
| 输入 | {  "type":"city ",  " parentCode ":"44"  } |
| 业务代码 | @Override  public List<SysDictEntity> selectDictByTypeAndParentCode(String type, String parentCode) {  return sysDictDao.selectDictByTypeAndParentCode(type, parentCode);  } |
| 输出 | {  "msg": "success",  "code": 0,  "dictList": [  {  "id": null,  "name": null,  "type": "city",  "parentCode": "44",  "code": "4401",  "value": "广州市",  "orderNum": null,  "remark": null,  "delFlag": null  },  {  "id": null,  "name": null,  "type": "city",  "parentCode": "44",  "code": "4402",  "value": "韶关市",  "orderNum": null,  "remark": null,  "delFlag": null  }…  ]  } |

### 5.8.2枚举查询设计

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.sys |
| 查询条件 | 枚举类型 |
| 列表显示 | 枚举类型、key值、value值 |
| 输入 | http://localhost:2320/hdi-server-admin/sys/dict/getEnumType/StatusEnum,IsGroupEnum |
| 业务代码 | @RequestMapping("/getEnumType/{type}")  public Hdi getEnumType(@PathVariable("type") String type) {  try {  String [] signs = type.split(",");  if(signs.length<1){  return Hdi.error("请传入正确的参数");  }  Map<String, Object> map = new HashMap<String, Object>();  for(String str : signs){  List<LinkedHashMap<String, Object>> list = new ArrayList<LinkedHashMap<String, Object>>();  Class clazz = Class.forName("com.ebig.hdi.common.enums."+str);  Method toKey = clazz.getMethod("getKey");  Method toName = clazz.getMethod("getValue");  //得到enum的所有实例  Object[] objs = clazz.getEnumConstants();  for (Object obj : objs) {  LinkedHashMap<String, Object> subMap = new LinkedHashMap<String, Object>();  subMap.put("key", toKey.invoke(obj));  subMap.put("value", toName.invoke(obj));  list.add(subMap);  }  map.put(str, list);  }  return Hdi.ok().put("list",map);  } catch (Exception e) {  throw new HdiException("字典查询出错" + e.getMessage());  }  } |
| 输出 | {  "msg": "success",  "code": 0,  "list": {  "StatusEnum": [  {  "key": 0,  "value": "停用"  },  {  "key": 1,  "value": "启用"  }  ],  "IsGroupEnum": [  {  "key": 0,  "value": "否"  },  {  "key": 1,  "value": "是"  }  ]  }  } |
|  |  |

### 5.8.3导入功能设计

### 5.8.4导出功能设计

## 5.9系统消息

### 5.9.1系统消息管理

#### 5.9.1.1分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 查询条件 | 消息类型 |
| 显示字段 | 消息类型、标题、简介、内容、状态、创建时间、修改时间、发布时间 |
| 排序 | 创建时间倒序 |
| 类包 | com.ebig.hdi.modules.sys |
| 输入 | {  "page":"1",  "limit":"10"  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  Page<SysMessageEntity> page = this.selectPage(new Query<SysMessageEntity>(params).getPage(),  new EntityWrapper<SysMessageEntity>().addFilterIfNeed(params.get(Constant.SQL\_FILTER) != null,  (String) params.get(Constant.SQL\_FILTER)).orderBy("create\_time", false));  return new PageUtils(page);  } |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 2,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id": 2,  "type": 0,  "title": "预警提醒消息",  "description": "预警提醒简介",  "content": "预警提醒测试内容",  "status": 0,  "createId": 1,  "createTime": "2019-04-26 15:47:10",  "editId": null,  "editTime": null,  "publishTime": null  },  {  "id": 1,  "type": 0,  "title": "系统消息",  "description": "测试消息简介",  "content": "系统消息测试内容",  "status": 0,  "createId": 1,  "createTime": "2019-04-26 15:46:00",  "editId": 1,  "editTime": "2019-04-26 15:59:33",  "publishTime": null  }  ]  }  } |

#### 5.9.1.2新增

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.sys |
| 输入 | {  "type": 0,  "title": "系统消息",  "description": "测试消息简介",  "content": "系统消息测试内容",  "status": 0,  ......  } |
| 业务代码 | public Hdi save(@RequestBody SysMessageEntity sysMessage) {  ValidatorUtils.validateEntity(sysMessage);  sysMessage.setCreateId(getUserId());  sysMessageService.save(sysMessage);  return Hdi.ok();  } |
| 输出 | {  "msg": "success",  "code": 0  } |

#### 5.9.1.3编辑

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.sys |
| 输入 | {  "id":1,  "type": 0,  "title": "系统消息",  "description": "测试消息简介",  "content": "系统消息测试内容",  "status": 0,  ......  } |
| 业务代码 | public Hdi update(@RequestBody SysMessageEntity sysMessage) {  ValidatorUtils.validateEntity(sysMessage);  sysMessage.setEditId(getUserId());  sysMessageService.update(sysMessage);  return Hdi.ok();  } |
| 输出 | {  "msg": "success",  "code": 0  } |

#### 5.9.1.4发布

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.sys |
| 输入 | {  "userIds" : [1,2,3,4,5],  "messageId" : "1"  } |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public void publish(Map<String, Object> params) {  List<Long> userIds = JSON.parseArray(params.get("userIds").toString(), Long.class);  if (StringUtil.isEmpty(userIds)) {  throw new HdiException("请选择发布用户");  }  Long messageId = Long.valueOf(params.get("messageId").toString());  SysMessageEntity sysMessageEntity = new SysMessageEntity();  sysMessageEntity.setId(messageId);  sysMessageEntity.setStatus(MessageStatusEnum.PUBLISH.getKey());  this.updateById(sysMessageEntity);  List<SysUserEntity> sysUserEntities = sysUserService.selectBatchIds(userIds);  for (SysUserEntity sysUserEntity : sysUserEntities) {  SysUserMessageEntity sysUserMessageEntity = new SysUserMessageEntity();  sysUserMessageEntity.setUserId(sysUserEntity.getUserId());  sysUserMessageEntity.setDeptId(String.valueOf(sysUserEntity.getDeptId()));  sysUserMessageEntity.setMessageId(messageId);  sysUserMessageEntity.setStatus(UserMessageStatusEnum.UNREAD.getKey());  sysUserMessageService.insert(sysUserMessageEntity);  }  } |
| 输出 | {  "msg": "success",  "code": 0  } |

#### 5.9.1.5查询详情

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.sys |
| 输入 | http://localhost:2320/hdi-server-admin/sys/sysmessage/1 |
| 业务代码 | public Hdi info(@PathVariable("id") Long id){  SysMessageEntity sysMessage = sysMessageService.selectById(id);  return Hdi.ok().put("sysMessage", sysMessage);  } |
| 输出 | {  "msg": "success",  "code": 0,  "sysMessageEntity ": {  "id":1,  "type": 0,  "title": "系统消息",  "description": "测试消息简介",  "content": "系统消息测试内容",  "status": 0,  ......  }  } |

### 5.9.2用户消息管理

#### 5.9.2.1分页查询

|  |  |
| --- | --- |
| 事项 | 内容 |
| 查询条件 | 消息类型 |
| 显示字段 | 消息类型、标题、简介、内容、创建时间、修改时间、发布时间、状态（阅读状态）、阅读时间 |
| 排序 | 创建时间倒序 |
| 类包 | com.ebig.hdi.modules.sys |
| 输入 | {  "page":"1",  "limit":"10"  } |
| 业务代码 | @Override  @DataFilter(subDept = true, user = false)  public PageUtils queryPage(Map<String, Object> params) {  int currPage = Integer.parseInt(params.get("page").toString());  int pageSize = Integer.parseInt(params.get("limit").toString());  Page<SysUserMessageEntityVo> page = new Page<SysUserMessageEntityVo>(currPage, pageSize);  List<SysUserMessageEntityVo> list = this.baseMapper.listForPage(page, params.get("userId"));  page.setRecords(list);  return new PageUtils(page);  } |
| 查询SQL | select u.\*,m.type,m.title,m.description,m.content from sys\_user\_message u left join sys\_message m on u.message\_id=m.id where u.user\_id=#{userId} and u.status!=2 |
| 输出 | {  "msg": "success",  "code": 0,  "page": {  "totalCount": 6,  "pageSize": 10,  "totalPage": 1,  "currPage": 1,  "list": [  {  "id":1,  "messageId": 1,  "deptId": 7,  "status": 0,  "readTime": "2019-04-28 09:11:11",  "type": 0,  "title": "系统消息",  "description": "测试消息简介",  "content": "系统消息测试内容",  ......  },  {  "id":2,  "messageId": 2,  "deptId": 7,  "status": 0,  "readTime": "2019-04-28 09:30:11",  "type": 0,  "title": "系统消息",  "description": "测试消息简介",  "content": "系统消息测试内容",  ......  ......  }......  ]  }  } |

#### 5.9.1.5查询详情

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.sys |
| 输入 | http://localhost:2320/hdi-server-admin/sys/sysusermessage/info/1 |
| 业务代码 | @Override  @Transactional(rollbackFor = Exception.class)  public SysUserMessageEntityVo queryInfoById(Long id) {  SysUserMessageEntity sysUserMessageEntity = new SysUserMessageEntity();  sysUserMessageEntity.setId(id);  sysUserMessageEntity.setStatus(UserMessageStatusEnum.READ.getKey());  sysUserMessageEntity.setReadTime(new Date());  this.updateById(sysUserMessageEntity);  return this.baseMapper.queryInfoById(id);  } |
| 查询SQL | select u.\*,m.type,m.title,m.description,m.content from sys\_user\_message u left join sys\_message m on u.message\_id=m.id where u.id=#{id} |
| 输出 | {  "msg": "success",  "code": 0,  "sysUserMessageEntityVo": {  "id":1,  "messageId": 1,  "deptId": 7,  "status": 0,  "readTime": "2019-04-28 09:11:11",  "type": 0,  "title": "系统消息",  "description": "测试消息简介",  "content": "系统消息测试内容",  ......  }  } |

#### 5.9.1.6删除

|  |  |
| --- | --- |
| 事项 | 内容 |
| 类包 | com.ebig.hdi.modules.sys |
| 功能说明 | 做软删除，更新状态操作 |
| 输入 | [1,2,3] |
| 业务代码 | @Override  public void delete(Long[] ids) {  for(Long id : ids) {  SysUserMessageEntity sysUserMessageEntity = new SysUserMessageEntity();  sysUserMessageEntity.setId(id);  //设置信息为删除状态  sysUserMessageEntity.setStatus(UserMessageStatusEnum.DELETE.getKey());  this.updateById(sysUserMessageEntity);  }  } |
| 输出 | {  "msg": "success",  "code": 0  } |