**长寿区督查“一张网”信息管理系统**

# 项目开发规范

南华中天信息技术有限公司

2018年08月03日

# 开发环境

Windows7、IntelliJ IDEA2017.3、Maven 3+、JDK1.8

## 服务器环境

服务器地址：192.168.0.242

账号：administrator

密码：admin@123

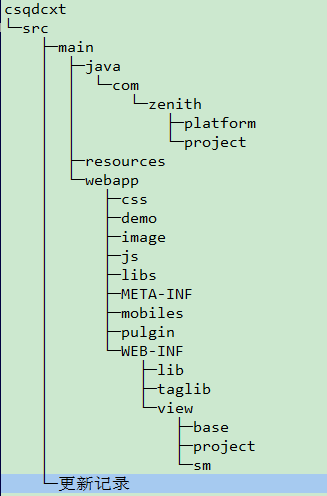
# 开发语言

目前仅限Java

# 编程规范

## 目录结构

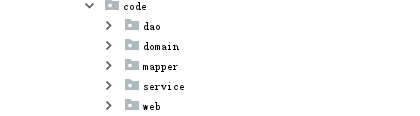
### 工程总目录结构



描述：

* src/main/com/zenith/platform 属于基础框架包，本身是以platform.jar引入使用【包含分页、主键生成，常规的工具包、上传附件功能、自动生成代码、后台权限等管理】。
* src/main/com/zenith/project 包是项目的业务模块汇总包，项目所涉及到的所有业务模块（除后台的权限模块），都写在此包内。
* src/main/resources约定除web.xml外，其余配置均存放在此目录中。
* webapp目录是存放和web项目相关的资源配置文件、页面文件等。
* 更新记录包里存放着项目数据库结构相关的文件或其他重要的升级文件。

#### JAVA类文件包目录结构



如：src/main/java/com/zenith/platform/sm/code

* dao包存放数据库操作类
* domain包存放数据实体类
* mapper包存放SQL的xml文件
* service包存放业务逻辑类
* web包存放http请求处理类

#### 配置文件目录结构



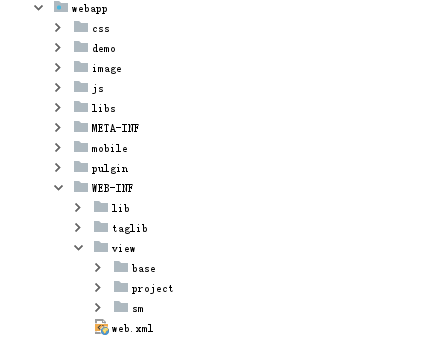
约定除WEB-INF/web.xml外，其余配置均存放在目录src/main/resources中。

描述：

以下配置在每个项目需要按需适时调整。

* resources/templates 目录是用于配置生成相应数据库表的实体包的模板。
* applicationContext.xml 和 spring-mvc.xml都是用于spring的配置
* file-manage.xml 用于上传文件功能的配置
* global.app.properties 资源地址或其他值的配置
* jdbc.properties 数据库连接地址的配置
* log4j.xml 日志输出的配置
* mybatis-config.xml 配置mybatis 持久化的属性。

#### webapp目录结构



描述：

* css 业务模块自定义的样式或者其它前端样式
* demo 上传附件的demo例子
* image 业务模块使用的图片
* js 业务模块自定义、或其它引入的小功能js
* libs 此包为QUI的内容，除非修改源代码bug，否则不建议往里面添加内容。
* mobile 给移动端使用，包含（css、js、html等）
* pulgin 引入独立的前端框架插件时，统一放到此包
* WEB-INF/lib 本项目所引用的jar
* WEB-INF/taglib 自定义jsp页面使用的特殊标签
* view 除移动端页面外，其余的jsp页面都在此包内
* view/base 一些基础报错提示页面
* view/project 业务模块的所有页面在此包下，按功能模块划分包名
* view/sm 后台权限模块的页面

## 命名规范

### 代码命名规范

1. Domain实体类

遵循Java类驼峰式命名规则，例如：数据库表名SYS\_CODE对应的实体类是SysCode.java。

实体类字段也采取驼峰式命名规则，通过去掉下划线，生成对应的字段名，例如：带下划线的level\_code对应的实体字段名是levelCode。

1. Dao数据库操作类

遵循Java类驼峰式命名规则，在实体类名称后追加上Dao作为数据库操作类名，例如：表SYS\_CODE的数据库操作接口类名为SysCodeDao.java。

注：dao层的类名需要同mapper包的xml文件保持一致，否则mybatis映射会出错。

1. Mapper XML文件

遵循Java类驼峰式命名规则，在实体类名后追加Dao作为xml文件的名称，例如：表SYS\_CODE的xml文件名称为SysCodeDao.xml。

注：dao层的类名需要同mapper包的xml文件保持一致，否则mybatis映射会出错。

1. Service业务逻辑类

遵循Java类驼峰式命名规则，在实体类名后追加Service作为业务逻辑接口类名，例如：表SYS\_CODE的接口名为SysCodeService.java，其实现类名为SysCodeServiceImpl.java。

1. web请求处理类

遵循Java类驼峰式命名规则，在实体类名后追加Controller作为请求处理类名，例如：表SYS\_CODE的请求处理类名为SysCodeController.java。

1. JSP页面

采用操作对象名加上对应的操作动作作为jsp文件的名称（全小写），例如：模块SysCode中，查询code列表数据为syscodelist.jsp；

其它例子：syscodeview.jsp【视图详情页面】

syscodeedit.jsp【基础信息编辑页面】

syscodeadd.jsp【基础信息新增页面】

syscodetree.jsp【基础信息树形展示页面或树形查询页面】

syscodetab.jsp。【tab选项卡页面】

### 数据库命名规范

#### 基本字符规则

采用26个英文字母（区分大小写）和0－9这十个自然数，加上下划线\_组成，共63个字符，不能出现其他字符（注释除外），我们要求所有对象不能超过30个字符。

|  |
| --- |
| Oracle：不区分大小写 |
| MySQL：表名区分大小写，字段不区分大小写 |
| SQL Server：不区分大小写 |

#### 表命名规则

实际名字尽量描述实体的内容，由**英文单词或单词组合**，**每个单词的首字母大写**，其他字母小写，不以数字和\_开头。

##### 基础规约

|  |  |
| --- | --- |
| **规则** | 模块名称缩写\_实际名称 |
| **缩写定义** | **模块名称前三位，或者复合词缩写** |
| **示例** | 如系统管理模块的机构管理  Sys——System 系统，Unit 机构  Sys\_Unit，代表系统管理的机构表 |

##### 业务主表

|  |  |
| --- | --- |
| **规则** | 业务模块名称缩写\_实际名称 |
| **缩写定义** | **模块名称前三位，或者复合词缩写** |
| **示例** | 收文表：Doc\_InDoc —— Incoming Document 缩写  发文表：Doc\_SendDoc ——Send Document 缩写 |

##### 业务从表

|  |  |
| --- | --- |
| **规则** | 模块名称缩写\_主表名称(或缩写)\_从表名称（名称太长用缩写） |
| **缩写定义** | **模块名称前三位，或者复合词缩写** |
| **示例** | 收文接收表：Doc\_InDoc\_Receive |

#### 字段命名规则

主键必须以“ID”命名

全部采用小写英文单词，单词之间用“\_”隔开；

如果该字段表示的是日期时间，则使用dtm为前缀；

如果该字段表示的是字符串，则使用chr为前缀；

如果该字段表示的是数字型，则用int为前缀；

oralce字段类型默认：string：nvarchar2；long、bigdecimal:number(x,y)(无精度数值对应long，有精度数值全部对应bigdecimal)；byte[]:blob；string (较长字符串):clob;

sqlserver字段类型默认：string：nvarchar；long、bigdecimal: numeric (x,y)(无精度数值对应long，有精度数值全部对应bigdecimal)；byte[]:img；string (较长字符串):text;

|  |  |
| --- | --- |
| **规则** | 实际名称 |
| **缩写定义** | 实际名称太长就用缩写 |
| **示例** | 主送单位：main\_send\_unit  创建时间：create\_time  接收日期：receive\_date  下载标志：download\_flag |

#### 视图命名规则

视图由几个表产生就用下划线（\_）连接几个表的缩写名，一定要列出所有表

|  |  |
| --- | --- |
| **规则** | 前缀\_各表名缩写\_实际名称 |
| **前缀定义** | v —— view首字母 |
| **示例** | 收文接收情况视图：v\_Doc\_InDoc\_Receive |

#### 序列命名规则

|  |  |
| --- | --- |
| **规则** | 前缀\_表名缩写 |
| **前缀定义** | seq —— sequence前3为字母 |
| **示例** | seq\_InDoc 收文表序列 |

#### 主键命名规则

|  |  |
| --- | --- |
| **规则** | 前缀\_表名缩写\_实际名称 |
| **前缀定义** | pk —— primary key缩写 |
| **示例** | pk\_InDoc\_ID —— 收文表主键 |

#### 外键命名规则

|  |  |
| --- | --- |
| **规则** | 前缀\_表名缩写\_实际名称\_外键名称 |
| **前缀定义** | fk —— forgein key缩写 |
| **完整名称定义** | fk\_InDoc\_Receive\_indocid 收文接收情况表的收文id外键 |

#### 索引命名规则

|  |  |
| --- | --- |
| **规则** | 前缀\_表名缩写\_实际名称 |
| **前缀定义** | idx —— index缩写 |
| **示例** | idx\_InDoc\_create\_time 收文表创建时间索引 |

#### 存储过程命名规则

|  |  |
| --- | --- |
| **规则** | 前缀\_模块名称缩写\_实际名称 |
| **前缀定义** | proc —— procedure缩写 |
| **示例** | proc\_Doc\_getInDocReceiveList 获取收文接收情况列表 |

#### 函数命名规则

|  |  |
| --- | --- |
| **规则** | 前缀\_模块名称缩写\_实际名称 |
| **前缀定义** | func —— function缩写 |
| **示例** | func\_Doc\_getInDocReceiveList获取收文接收情况列表 |

#### 触发器命名规则

|  |  |
| --- | --- |
| **规则** | 前缀\_模块名称缩写[\_表名缩写]\_实际名称 |
| **前缀定义** | tri —— trigger缩写 |
| **示例** | tri\_InDoc\_download\_update 收文下载更新 |

#### 表空间命名规则

|  |  |
| --- | --- |
| **规则** | 前缀\_模块名称缩写\_实际名称【自动生成】 |
| **前缀定义** | ts—— tablespace缩写 |
| **示例** | ts\_Doc\_InDoc 收文表空间 |

临时表空间：结尾加\_tmp

分区表空间：结尾加\_pn

### 其他文件命名规范

## 注释规范

### 代码注释

1. 类的注释：写出该类的作用，修改人和修改时间；例如：

*/\*\*  
 \* 人员组管理Controlller*

*\*  
 \** ***@data*** *2018-08-06*

*\** ***@author*** *zhanghl  
 \*/*

1. 方法的注释：写出该方法的作用，参数描述，修改人和修改时间；例如：

*/\*\*  
 \* 根据Groupid加载用户组树  
 \*   
 \** ***@param groupIds*** *\* 组的id，可以是多个以逗号分开  
 \** ***@return*** *用户组的树  
 \** ***@data 2018-08-06*** *\** ***@author*** *zhanghl  
 \*/*

1. Js的注释：同java方法的注释一致。
2. 实体类的注释：每个字段需要明确的注释描述。

### 数据库注释

1. 数据库设计中涉及所有库、表、字段、视图、序列、主键、外键、索引、存储过程、函数、触发器等都必须添加中文注释；
2. 针对表中设计的标志位，必须标明可能出现的标志的中文含义或该标志位的数据来源；
3. 针对表中涉及其他表的主键或其他字段，必须标名相关表对于字段；
4. 针对存储过程、函数的参数，必须标明其含义。