**项目开发规范V1.1**

版本：V1.0

时间：2019年04月16日

为了团队开发过程中代码的可读性，可维护性，优雅性，降低代码耦合度等，提升团队开发效率，降低团队沟通成本，特此规定本公司内部Java项目开发规范，此开发规范是基于阿里巴巴Java开发手册v1.3.0版本，在阿里巴巴Java开发手册的基础上做适当的补充，阿里巴巴Java开发手册中已说明的，不在此开发规范中重复提出。

1. **项目结构**
2. **包定义**

所有公司内部软件开发，package都以com.tax为前缀。

1. **MVC架构**

严格遵循MVC设计模式，分成架构，各个层次的包如下定义：

控制器层类：com.tax.\*.\*.controller，以Controller为后缀

服务层接口：com.tax.\*.\*.service，以Service为后缀

服务层实现类：com.tax.\*.\*.service.impl，以ServiceImpl为后缀

通用业务实现类：com.tax.\*.\*.manager，以Manager为后缀

配置类：com.tax.\*.\*.config，以Config为后缀

持久层接口：com.tax.\*.\*.mapper，以Mapper为后缀（使用MyBatis）

持久层实现：resources/mapper/\*/\*，以mapper.xml为后缀（使用MyBatis）

数据对象类：com.tax.\*.\*.bean，以Bean为后缀，通过Bean层向上传输数据源对象

数据传输对象类：com.tax.\*.\*.dto，以DTO为后缀，service和manager向外传输的对象（非必须类，如无必要可不创建，但出现大量无关字段，不许污染原有Bean，只能以dto为对象进行传输）

值对象类：com.tax.\*.\*.vo，以VO为后缀，向视图层传输的对象（非必须类，如无必要可不创建，但出现大量与原Bean无关字段，不许污染原有Bean，只能以dto为对象进行传输）

通用工具类：com.tax.\*.\*.common，以Utils为后缀

常量类：com.tax.constant，以Constants为后缀

枚举类：com.tax.enums，以Enum为后缀

过滤器类：com.tax.config.filter，以Filter为后缀

监听器类：com.tax.config.listener，以Listener为后缀

网络操作类：com.tax.net，服务端以Server为后缀，客户端以Client为后缀

异常类：com.tax.exception，以Exception为后缀

注解类：com.tax.annotation

切面类：com.tax.aspect，以Aspect为后缀，横切关注点一定要加注释

1. **目录结构**

文档目录：项目根目录中必须增加documents目录，用于存储项目整个开发过程中涉及的所有文档，包括需求文档，数据库文档（每次版本更新，数据库结构的执行脚本一定要留存），详细设计文档，用户手册等。

框架配置文件目录：在src/resources根目录中创建config目录用于存储所有框架相关的配置文件。

映射文件目录：在src根目录中创建mapper目录用于存储MyBatis的Mapper文件。

如果使用SpringBoot，则默认按照SpringBoot的目录存储规则，为了便于系统升级等操作，可以选择独立的目录来存储上传文件。

1. **访问URL**

使用Restful API接口的形式，绝大部分URL都要使用Restful API接口形式，少部分可以传递查询字符串，视具体情况而定，枚举类型数据一定要存储于字典模块，严禁出现以汉字作为固定内容查询。

1. **文档要求**
2. **数据库文档**

数据库文档必须以.sql结尾，包含创建数据库，创建表，创建约束，初始化表数据的所有SQL语句。并且调整好建表的顺序，所有的名称避免出现数据库关键字，任何名称不要用`号括起来。

1. **需求文档**

需求文档随时要保持到最新状态，如果有需求的变更，要及时更新需求文档。

1. **详细设计文档**

详细的实现需要体现在文档上，并及时更新。

1. **数据库表字段设计文档**

当数据库有更新时，表字段设计文档也要及时更新，并备注。

补充：列表数据全部使用逻辑删除，并记录更新人与时间，数据库只有删除关联关系这一种情况可执行delete语句删除，尽量避免批量操作语句，更不要出现锁表操作。

1. **异常与日志记录**
2. **什么时候记录日志**

当出现异常时，当重要操作时如修改或删除数据，登录操作，权限分配等。

出现异常保存error级别的日志，操作记录保存info级别的日志。

1. **日志周期**

日志以天为时间间隔保存，一天对应一个日志文件，日志文件的最短保留时间设置为15天。

1. **自定义异常**

ServiceException：应用异常类

DAOException：数据库异常类，继承自ServiceException

ServiceException：服务异常类的父类，继承自ServiceException

1. **异常处理机制**

由于使用了框架的项目，异常被封装成RuntimeException的子类重新抛出，故可以不捕捉任何异常，最终交由框架统一处理异常信息，但是为了更好地做异常日志记录，在每一层中加上try...catch...语句块，以切面环绕通知方式，记录入参，时间，方法，异常栈等信息。

未使用框架的项目：DAO层捕捉异常，抛出DAOException，Service层捕捉DAOException并重新抛出ServiceException，控制器层捕捉ServiceException并转发到统一框架异常的错误处理hander；

1. **项目管理**

**1）项目管理工具**

项目使用Maven进行管理，所有第三方JAR库都使用最新稳定版本。使用过程中注意排除JAR库的版本冲突。

1. **协同开发**
2. **代码更新（Update/Pull）**

每次版本开发必须从主支迁出分支，在分支版本进行更新，如果分支版本开发功能稳定，上线后，完全解决完冲突由模块开发相关人员合并分支到主干。

每天第一件事情更新代码，提交代码前也更新代码，如果出现代码冲突，则考虑清楚如何合并代码（新人入职一个月内出现冲突，必须要通知到冲突影响代码者）。

1. **代码提交（Commit/Push）**

每天提交一次代码，如果功能未完成，提交至本地版本，功能部分完成，提交至分支。

提交代码必须写明提交信息，提交信息禁止出现：更新，修改bug，修改功能等类似的简单描述，一定要具体说明做了哪些事情，每一件事情按序号标出，每一个序号的内容结束必须换行。

代码提交前做周全的单元测试，确定没问题才能提交。

及时提交确认无误可运行的代码，不要等到所有代码都写完了再提交。

1. **.gitignore文件**

使用Git，则必须增加.gitignore文件，把不需要添加到版本控制的目录排除。

1. **哪些文件需要提交**

与项目源代码无关的文件不要提交到版本控制仓库中，能提交的代码通常限定于Java源代码，html页面，CSS样式，JS脚本，文档等。编译后生成的class等文件一律禁止提交。项目更新下来后，由IDE生成或修改的配置文件，视具体情况而提交。

1. **代码编写**
2. **泛型**

需要用到泛型的地方，则使用泛型，并灵活运用extends和super关键字。

1. **序列化**

需要序列化的类，必须增加序列化版本号，由IDE生成。

1. **代码警告**

杜绝出现代码警告，如果警告无法通过代码优化清除掉，则使用压制警告，并且注意压制警告的有效位置。

# 规范要求安装 Alibaba Java Coding Guidelines扫码代码，eclipse同。

1. **单元测试**

重要方法必须要做单元测试，使用JUnit或Spring JUnit做测试，测试类所在的包名称与源代码的包名称一致，测试类以Test为后缀，测试方法以test为前缀，测试的方法名为后缀。

1. **前端开发**

移动端页面中，样式的引入在<head>标签内部，JavaScript文件的引入在页面的尾部，使用外部样式和外部JS，少部分情况可以直接使用内部样式和内部JS。外部静态文件的引入不使用相对路径的方式。移动端开发，必须在统一文件中设定no-cache处理并引入，移动端页面调用，增加后缀&\_=时间戳。

1. **前端页面、静态文件命名**

html页面的名称、静态文件的名称，如果是多个单词，则每个单词用\_连接，严禁使用中文。

1. **CSS选择器**

CSS样式文件中的类或id，如果是多个单词，则每个单词用-连接。

1. **HTML标签id**

页面中HTML标签的id，如果是多个单词，则每个单词用-连接。

1. **枚举**

表示状态的枚举，表示性别的枚举，表示是与否的枚举等，能用枚举则不用常量。

**10.MyBatis mapper映射文件命名**

MyBatis mapper映射文件以DAO的简单类名开头，以.xml结尾。

1. **字段类型**

对于字段类型来说，能用基础类型则用基础类型，如果数据库必须存在null的情况则使用包装类，字段选取应用根据情况而定。

系统中凡是涉及金额运算：全部用BigDecimal，BigDecimal构造时一律用String构造（原因自行百度），实体里金额字段声明非必填项用Double，必填项用double，次之替代为String，参数传递也相同。数据库层面全部用DECIMAL(10,2)，尽量避免前端与数据库层面的二级运算，凡涉及金额无特殊要求一律四舍五入保留两位，多次参与运算的金额在最后再保留。

1. **第三方库/类使用**
2. **字符串处理**

使用commons-lang核心包的StringUtils类

1. **Java Bean拷贝**

使用Apache Dozer库

1. **JSON处理**

使用阿里巴巴com.fasterxml.jackson库

1. **文件处理**

使用commons-io包的FileUtils类

1. **数据源与数据连接池**

强制使用阿里巴巴的druid库，对于连接，必须建立池化管理思想

1. **Office文档处理**

使用Apache POI库

1. **HTTP网络操作**

使用Apache HttpClient库

1. **NIO/Socket**

使用JBoss提供的Netty或Apache的Mina，凡涉及io操作，未继承AutoCloseable接口的，必须在try（）内开启，或者在finally中去关闭流，避免出现异常时，文件占用阻塞流程正常工作。