### 短信平台开发各规范

1. 各模块划分
   1. Façade：包含业务模型和接口定义，仅供业务端使用。
   2. Service：对业务端接口定义的具体实现，调用Engine时将业务模型转换为核心模型，返回业务端时把核心模型转换成业务模型。

注意：不涉及核心业务。

* 1. Engine：包含核心模型和核心业务实现以及Dao的接口定义。供Service及内部调用，不需要定义接口。直接写实现。

注意：只处理内部模块，不依赖其他任何模块。

* 1. Dao：包含数据库模型和Engine的实现。将核心模型转换为数据库模型调用。并将数据库模型转换成核心模型返回。

1. 命名规范
   1. Façade：业务模型以Bean结尾，接口以I开头以Service结尾。
   2. Service：接口实现以ServiceImpl结尾。
   3. Engine：核心模型以Entity结尾，接口实现以Service结尾，接口定义以I开头以Service结尾。
   4. Dao：数据库模型需跟表明保持一致。
2. 包名规范
   1. 文件按照模块划分，强关联关系可以放到同一个路径下，如：

com.pzj.core.smp.record.recordModel.class

com.pzj.core.smp.record.IRecordService.class

1. 代码规范
   1. Eclipse中配置统一的spring-checkstyle模板，见附件。
   2. 类名、方法名、变量名采用英文定义，复数采用s结尾。禁止使用拼音、变量类型作为变量名及后缀。

如：List、Set、Map，当定义一个类型List<Object>时，变量名为：names、orders，不允许使用nameList、orderList。

* 1. JAVA源文件除去import、空行等内容，代码行数控制在300行范围内。做到2-3屏能拉到文件底部。
  2. 禁止使用System.out.println()打印。
  3. 项目中加入doc目录存放各类型文档。

1. 异常处理
   1. wiki地址：<http://wiki.mftour.cn/pages/viewpage.action?pageId=7443347>
2. 日志规范
   1. wiki地址：<http://wiki.mftour.cn/pages/viewpage.action?pageId=6916399>
3. Git使用规范
   1. wiki地址：<http://wiki.mftour.cn/pages/viewpage.action?pageId=6359231>
4. 数据库开发规范
   1. wiki地址：<http://10.0.18.11/pages/viewpage.action?pageId=3768509>

9 附件