## 桌面闹钟开源代码的质量分析报告

通过初步阅读分析其代码质量较高，故对其进行进一步的质量分析和审查。

### 代码质量分析方法

代码质量分析主要由人工分析和利用工具分析两种方式，二者各有优缺点，在实际过程中需要结合使用以保证正确性、发挥效能。

1. **人工**

通过人工发现代码中存在的缺陷和质量问题是分析软件质量最直接的手段。软件开发相关人员在软件实现、测试、维护过程中，主动发现编写代码中存在的问题并给予修改，但是分析代码的效率低，寻找缺陷不够全面，比如其中软件的深层次问题可能难以发现。

1. **自动**

利用SonarQube，FindBugs等工具进行代码质量分析，从程序的语法、结构、接口等方面进行代码审查，并能够对代码风格进行分析。其优点是代码分析效率高，且能够发现软件开发人员忽略的错误，在实际应用中要结合人工审查才能完全明确代码缺陷。

本报告主要使用SonarLint工具进行针对小米便签开源软件进行质量分析，辅以少量的人工分析。

### Sonar及SonarQube简介

SonarQube软件是一种静态代码检查工具，采用B/S架构，帮助检查代码缺陷，改善代码质量，提高开发速度，通过插件形式，可以支持Java、C、C++、JavaScript等等二十几种编程语言的代码质量管理与检测。

Sonar客户端可以采用IDE插件、Sonar-Scanner插件、Ant插件和Maven插件方式，并通过各种不同的分析机制对项目源代码进行分析和扫描，并把分析扫描后的结果上传到Sonar的数据库，通过Sonar web界面对分析结果进行管理。

### 使用SonarLint进行分析

代码中的缺陷分为如下几个等级：

1. Blocker：极有可能影响应用程序表现的错误；
2. Critical：可能影响应用程序表现的错误和表示安全缺陷的问题；
3. Major：严重影响开发者效率的质量缺陷；
4. Minor：轻微影响开发者效率的质量缺陷；
5. Info：不是错误或者质量缺陷。

（1）Blocker

1. 在if/else/for/while/do语句中必须使用大括号，即使只有一行代码，避免使用下面的形式：if (condition) statements;
2. 加@Override注解。

3、线程池不允许使用Executors去创建，而是通过ThreadPoolExecutor的方式，这样的处理方式让写的同学更加明确线程池的运行规则，规避资源耗尽的风险。

（2）Critical

1、Object的equals方法容易抛空指针异常。

2、不能使用过时的类或方法。

3、在一个switch块内，每个case要么通过break/return等来终止，要么注释说明程序将继续执行到哪一个case为止；在一个switch块内，都必须包含一个default语句并且放在最后，即使它什么代码也没有。

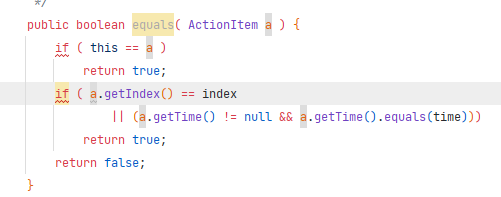
1. 所有编程相关的命名均不能以下划线或美元符号开始。在接口中定义常量会要求其子类必须含有这些常数，原则上应该将此接口中常数删去或将其接口变为final类型。

4、方法名、参数名、成员变量、局部变量都统一使用lowerCamelCase，必须遵从驼峰形式。

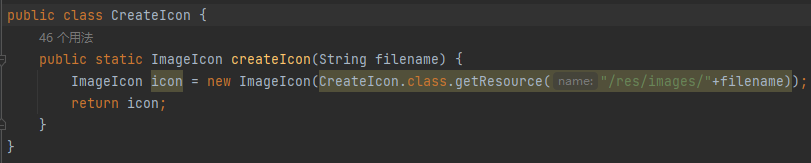
5、线程资源必须通过线程池提供，不允许在应用中自行显式创建线程。

（3）Major

1. 不允许未经定义的常量直接出现在代码中。
2. 类名使用UpperCamelCase风格，必须遵从驼峰形式，但以下情形例外：（领域模型的相关命名）DO / BO / DTO / VO / DAO。
3. Math.random() 这个方法返回是double类型，注意取值的范围[0,1)。
4. Sections of code should not be commented out：不需要被使用的代码段不需要被注释掉，应该直接删除。
5. 中括号是数组类型的一部分，数组定义如下：String[] args。
6. 类、类属性、类方法的注释必须使用javadoc规范，
7. 不要在条件判断中执行复杂的语句，将复杂逻辑判断的结果赋值给一个有意义的布尔变量，以提高可读性。



1. 工具类应该隐藏public构造器



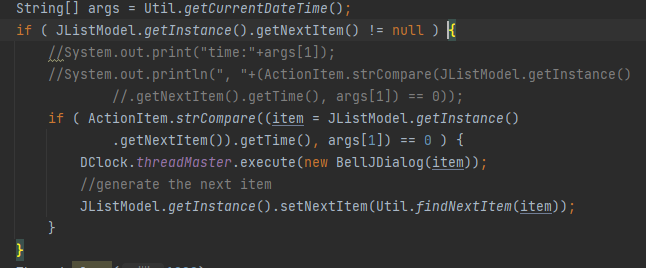
1. invokes inefficient new String(String) constructor

调用低效的new String(String)构造函数



10、避免实例化字符串对象；这通常是不必要的。



11、嵌套的if语句可以组合起来

12、 去除空的if语句



13、 去除多余修饰符： 'public'。



14、 表达式可被简化

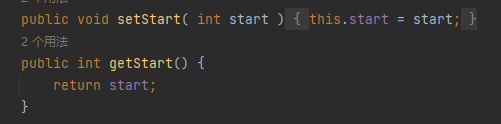


15、 Avoid using dollar signs in variable/method/class/interface names

避免在变量/方法/类/接口名称中使用美元符号

图形用户界面, 文本, 应用程序

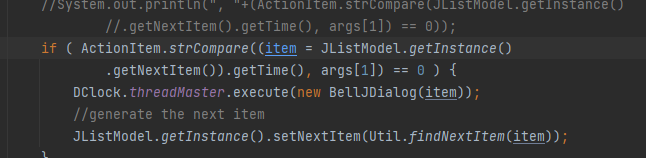
描述已自动生成



16、避免使用没有大括号的if...else语句.



17、 应避免在子表达式中赋值



18、属性和方法声明应位于内部类之前

文本

描述已自动生成