**旅行打卡小程序**

****

**课 程： 软件工程**

**题 目：基于微信小程序的旅行打卡设计与开发**

**文档主题： 代码走查报告书**

**专 业： 软件工程**

**班 级：** 1801

制作人员： SE2020-G7小组

**文档修订记录**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 编号 | 修改章节名称 | 说明 | 修订日期 | 版本号 | 修订人 | 发布日期 |
| 01 | 全文 | 新建 | 2020.12.16 | v0.1.0 | 王心怡 | 2020.12.17 |
| 02 | 全文 | 修订 | 2020.12.26 | v0.2.0 | 王心怡 | 2020.12.28 |
| 03 | 全文 | 填写报告 | 2020.12.28 | V0.3.0 | 吴卓、闫紫微 | 2020.12.28 |

目录

[目录 3](#_Toc520)

[代码走查报告书 4](#_Toc8136)

[1. 引言 4](#_Toc13199)

[1.1编写目的 4](#_Toc1364)

[2.走查明细 4](#_Toc11302)

[3.参考资料 6](#_Toc22780)

# 代码走查报告书

# 引言

## 1.1编写目的

本文档为G07小组为有效规范代码编写质量编写的代码走查报告。[2]

# 2.走查明细

|  |  |  |
| --- | --- | --- |
| **审查对象：“足之迹”微信小程序源代码** | | |
| **审查人：闫紫微、吴卓** | | |
| **审查日期：2020-12-28** | | |
| **审查项** | **审查结果** | **是否通过** |
| **准备工作** | | |
| 1.一份解释代码的最新的详细设计文档 | √ | 通过 |
| 2.代码已提交至github，并标明版本号 | √ | 通过 |
| **模块接口** | | |
| 1.参数的数目与变元是否一致 | √ | 通过 |
| 2.参数的次序与变元是否一致 | √ | 通过 |
| 3.参数的属性与变元是否一致 | √ | 通过 |
| 4.参数的单位系统与变元是否一致 | √ | 通过 |
| 5.是否修改了只作输入用的变元 | √ | 通过 |
| 6.全局变量的定义在各个模块中是否一致 | √ | 通过 |
| 7.全局变量的用法在各个模块中是否一致 | √ | 通过 |
| **局部数据结构** | | |
| 1.局部数据是否进行了说明 | √ | 通过 |
| 2.局部数据是否进行了初始化、是否有默认值 | √ | 通过 |
| **重要的执行通路** | | |
| 1.是否考虑了最有代表性的执行通路 | √ | 通过 |
| 2.是否考虑了最可能发现错误的执行通路 | √ | 通过 |
| **出错处理通路** | | |
| 1.是否设置了适当的处理错误的通路 | √ | 通过 |
| **边界条件** | | |
| 1.是否考虑了边界条件 | √ | 通过 |
| **功能** | | |
| 1.是否可以实现详细设计说明中描述的功能 | √ | 通过 |
| 2.功能的完成度如何 | **95%** | 通过 |
| **性能** | | |
| 1.是否可以达到详细设计说明中描述的性能 | √ | 通过 |
| 2.性能的程度如何 | **90%** | 通过 |
| **页面** | | |
| 1.页面是否和界面设计的样式一致 | **90%** | 通过 |
| 2.如有改动，效果如何 | **很好** | 通过 |
| **编码规范** | | |
| 1.WXML中各种组件的class属性命名格式是否符合G07的编码规范 | √ | 通过 |
| 2.WXML中各种组件的缩进格式是否符合G07的编码规范 | √ | 通过 |
| 3.WXSS中多种属性的排列格式是否符合G07的编码规范 | √ | 通过 |
| 4.WXSS中多种属性的缩进格式是否符合G07的编码规范 | √ | 通过 |
| 5.JS中多种变量的命名格式是否符合G07的编码规范 | √ | 通过 |
| 6.页面的命名格式是否符合G07的编码规范 | √ | 通过 |
| 7.云函数的命名格式是否符合G07的编码规范 | √ | 通过 |
| 8.云数据库中集合和其中字段命名格式是否符合G07的编码规范 | √ | 通过 |
| **条件判断** | | |
| 1.if-else结构使用是否正确 | √ | 通过 |
| 2.判断体是否足够短且清晰 | √ | 通过 |
| 3.判断次数是否足够小 | √ | 通过 |
| 4.是否考虑了所有情况，并先判断最常用情况 | √ | 通过 |
| **循环** | | |
| 1.循环前是否完成相应初始化 | √ | 通过 |
| 2.循环控制条件是否清晰正确 | √ | 通过 |
| 3.循环终止条件是否正确 | √ | 通过 |
| 4.循环体结构是否清晰 | √ | 通过 |
| 5.循环嵌套是否尽量小于三层 | √ | 通过 |
| 6.是否存在无穷尽循环 | √ | 通过 |
| 7.循环中涉及的数组、集合下标是否越界 | √ | 通过 |
| **云函数** | | |
| 1.云函数是否高内聚 | √ | 通过 |
| 2.云函数是否和其他代码松耦合 | √ | 通过 |
| 3.云函数是否正确初始化 | √ | 通过 |
| 4.云函数的返回值是否正确定义 | √ | 通过 |
| 5.调用云函数时，名称和传递数据是否准确 | √ | 通过 |
| 6.云函数异常是否可以处理清除 | √ | 通过 |
| **注释**[1] | | |
| 1.所有注释是否为最新的 | √ | 通过 |
| 2.所有注释是否清楚且正确 | √ | 通过 |
| 3.WXML中每个组件是否有注释说明是哪块结构 | √ | 通过 |
| 4.WXSS中每个属性是否说明了属于哪个组件 | √ | 通过 |
| 5.JS中是否对每个函数、变量注释了功能 | √ | 通过 |
| **程序组织结构**[2] | | |
| 1.所有代码是否结构清晰，具有良好的结构外观 | √ | 通过 |
| 2.所有模块是否定义清晰，模块分解清楚 | √ | 通过 |
| 3.所有接口是否模块化，修改时是否不影响其他代码模块 | √ | 通过 |
| 4.数据库操作、IO操作等是否正确关闭资源 | √ | 通过 |
| 5.是否有无效的代码或者无用的代码 | √ | 通过 |

# 3.参考资料

[1]代码走查报告（模板）（参考于2020.12.26）

[EB/OL].https://wenku.baidu.com/view/93aa740cba1aa8114431d985.html

[2]软件测试胡侃之代码审查和代码走查（参考于2020.12.26）

[EB/OL].https://blog.csdn.net/weixin\_31484477/article/details/81074198