**源石智影iOS小组**

**代码规范 试行版**

**版本 1.0**

**修订历史**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** | **参与人员** |
| 2017年4月5日 | 1.0 | 试行版 | 张一奇 | 王强、白召雷 |

备注：iOS代码规范1.0版本是试行版，今后会根据项目情况做相应调整

**源石智影 iOS开发团队**

目录

[1、简介 3](#_Toc479177377)

[2、目的 3](#_Toc479177378)

[3、适用范围 3](#_Toc479177379)

[4、iOS通用规则 3](#_Toc479177380)

[4.1 下面所有规则对第三方类库无约束 3](#_Toc479177381)

[4.2 命名规范 3](#_Toc479177382)

[4.3 通用规范 3](#_Toc479177383)

[5、智影iOS小组代码规范 4](#_Toc479177384)

[5.1 命名规范 4](#_Toc479177385)

[5.2 资源管理 4](#_Toc479177386)

[5.3 代码习惯 4](#_Toc479177387)

[5.4 代码结构 5](#_Toc479177388)

[5.5 代码重构 5](#_Toc479177389)

[5.6 代码提交 5](#_Toc479177390)

## 1、简介

iOS代码规范具体到代码细节上，比如命名，注释，代码间隔。使iOS团队写出规范和风格统一的代码

## 2、目的

使智影iOS团队写出的代码规范，可读性高，风格统一。

## 3、适用范围

iOS代码规范适用于由小组开发的基于swift语言的项目。本文档将供源石智影iOS开发团队使用。

## 4、iOS通用规则

### 4.1 下面所有规则对第三方类库无约束

\* 所有类、方法、属性等命名，使用英文，不要使用拼音，做到见名知意，采用驼峰式命名规则

\* 根据资源类型或者所属业务逻辑对项目资源进行分组，使得整个项目结构清晰明了

\* 整个项目保持一种代码书写风格，让你的代码变的优雅！

### 4.2 命名规范

\* 所有类名称以项目工程开头命名

\* 属性名使用小驼峰命名法

\* 项目中所有Group或者文件名称（图片名字等），不要使用汉字命名，要使用英文命名

\* 整个项目保持一种代码书写风格，让你的代码变的优雅！

### 4.3 通用规范

\* 基于MVVM设计模式原则，至少要保证controller与数据处理，网络请求相对独立

\* 请多使用 #pragma mark - 或 //MARK: -  对方法进行分组

## 5、智影iOS小组代码规范

### 5.1 命名规范

\* 类名称使用大驼峰命名法，以“ZY”开头，eg：“ZYView”

\* 属性名使用小驼峰命名法

\* 开放属性命名前需要加上模块名和下划线，方便他人调用，

eg: “ var chat\_Model: ZYChatModel?”

\* 全局变量命名前需要加上模块名和下划线，防止冲突

### 5.2 资源管理

\* 项目project中的文件夹名称和位置，要和Finder中真实名称与位置相对应

\* 一般将图片放入Assets.xcassets资源库中

\* 在Assets.xcassets中，需要按模块建立文件夹，图片命名前需要加上模块名和下划线，

eg：“chat\_delete.png” 和 “message\_delete.png” 虽然是同一张图片，但命名不同，所处不同的文件夹，不会导致冲突

\* 对于不常用的图片，直接放到对应模块的“Other”文件夹下

eg：新特性页面的图片只有在app更新的时候才会使用，请将图片放到文件夹里，降低内存占用

### 5.3 代码习惯

\* 删除多余的空行

eg：1.所有方法与方法之间空1行

2.所有代码块之间空1行

\* 删除多余的注释

eg：1.注释掉的代码如果一周以上不再使用，请删除

2.删除没有意义的注释，保证页面简洁

\* 删除多余的方法

eg：1.如果方法没有使用到，请删除它

2.如果方法没有执行任何业务逻辑，请删除它或者给出一定注释

\* 删除未被使用的资源文件

\* 添加必要的注释

eg：1.所有自定义属性 需要给出注释

2.所有自定义的方法需要给出注释

3.比较大的代码块需要给出注释

4.程序中出现加密／解密 逻辑的操作地方，需要给出注释说明过程（无论是系统还是自定义）

\* 整体代码风格需要统一

eg：1. 代码后面的”{“ 不需要单独占用一行

2. //MARK: - 与下面的代码之前不要空行

3. swift的枚举值不需要写枚举名，eg：“ .normal ”

### 5.4 代码结构

\* 一个文件中的代码应遵守以下顺序 “开放属性 -> 私有属性 -> deinit {} -> 开放方法 -> 私有方法 -> 懒加载控件”

eg： “ var chat\_Model: ZYChatModel?

private lazy var iconImage = UIImageView()

deinit {}

func sendMessage() {}

private func setupUI() {}

private lazy var sendBtn: UIButton = UIButton()

”

\* 代理方法应写到扩展里

eg： extension { // code }

\* 如果在extension 中使用本类属性或方法，需将 private 改为 filePrivate

### 5.5 代码重构

\* 单页代码行数要求在1000行以内，每个方法最好不要超过100行，单页代码过多难以维护，如果单页代码超过1000行，请对代码进行重构

eg：“使用单例抽取聊天控制器的录音机模块，放到另外一个文件中，只提供录音和播放的接口”

### 5.6 代码提交

\* 每天上班前拉取最新代码，下班前将代码提交

\* 勤提交代码，每完成一个功能模块请在SourceTree上提交，防止过久不提交产生的代码冲突