**代码评审标准**  
1. Code Review目的   
Code Review是一种用来确认方案设计和代码实现的质量保证机制，通过这个机制我们可以对 代码、测试过程 和 注释 进行检查。   
Code Review主要用来在软件工程过程中改进代码质量，通过Code Review可以达到如下目的：   
在项目早期就能够发现代码中的BUG。   
帮助初级开发人员学习高级开发人员的经验，达到知识共享。 避免开发人员犯一些很常见，很普通的错误。 保证项目组人员的良好沟通。 项目或产品的代码更容易维护。  
  
2. Code Review的前提条件   
代码提交审核前，开发者 必须确保代码符合如下条件，审核者 需要确保所有前提条件都已满足方可开始审查，同时也是审查的主要检查点。   
所有代码注释清晰，语法正确，编译通过。   
日志代码完整，业务日志、系统日志分开，中文描述，脱敏处理，状态变更，全部清晰明确。   
 测试代码覆盖全部分支和流程，暂时统一使用工具Emma（各编译器可下载对应插件）进行Coverage Check。   
项目引用关系明确，依赖关系清晰，配置文件描述。   
  
3. Code Review的审查范围   
代码的一致性、编码风格、代码的安全问题、脱敏问题、代码冗余、是否正确设计以符合设计要求（性能、功能）与设计文档相同等等。   
3.1、 完整性检查（Completeness）   
代码是否完全实现了设计文档中所涉及的所有流程和功能点   
代码是否已包含所有所需的业务日志、系统日志、异常日志，日志内容是否完整，日志文件配置是否正确。   
代码是否使用缓存等，配置信息是否正确可配置。   
代码中是否存在任何没有定义或没有引用到的变量、常数或数据类型等   
3.2、 一致性检查（Consistency）   
代码的逻辑是否符合设计文档   
代码中使用的格式、符号、结构等风格是否保持一致   
3.3、 正确性检查（Correctness）   
代码是否符合制定的标准 所有的变量都被正确定义和使用 所有的注释都是准确的   
所有的程序调用都使用了正确的参数个数   
3.4、 可修改性检查（Modifiability）   
代码涉及到的常量是否易于修改(如使用配置、定义为类常量、使用专门的常量类等)   
代码中是否包含了交叉说明或数据字典，以描述程序是如何对变量和常量进行访问的。  
代码是否只有一个出口和一个入口（严重的异常处理除外）   
3.5、 可预测性检查（Predictability）   
代码所用的开发语言是否具有定义良好的语法和语义 是否代码避免了依赖于开发语言缺省提供的功能 代码是否无意中陷入了死循环 代码是否避免了无穷递归   
3.6、 健壮性检查（Robustness）   
代码是否采取措施避免运行时错误（如数组边界溢出、被零除、值越界、堆栈溢出等）  
3.7、 结构性检查（Structuredness）   
程序的每个功能是否都作为一个可辩识的代码块存在 循环是否只有一个入口   
3.8、 可追溯性检查（Traceability）   
代码是否对每个程序进行了唯一标识   
是否有一个交叉引用的框架可以用来在代码和开发文档之间相互对应 代码是否包括一个修订历史记录，记录中对代码的修改和原因都有记录 是否所有的安全功能都有标识   
3.9、 可理解性检查（Understandability）   
注释是否足够清晰的描述每个子程序   
是否使用到不明确或不必要的复杂代码，它们是否被清楚的注释 使用一些统一的格式化技巧（如缩进、空白等）用来增强代码的清晰度 是否在定义命名规则时采用了便于记忆，反映类型等方法 每个变量都定义了合法的取值范围   
代码中的算法是否符合开发文档中描述的数学模型   
3.10、 可验证性检查(Verifiability)   
代码中的实现技术是否便于测试 测试代码是否正确，是否覆盖所有流程

4. Code Review的步骤   
目前Code Review 步骤暂定如下，试行一段时间再根据问题做调整。   
4.1、代码编写者 和 代码审核者 坐在一起，由 代码编写者 按照设计文档中的用例依次讲解自己所写的代码和相关逻辑，可采用从前端到后台的方式，例如从Web层->DAO层。   
4.2、 代码审核者 在此过程中可以随时提出自己的疑问，同时积极发现隐藏的bug；代码编写者 和 代码审核者 都要对这些bug记录在案，代码编写者 修改后再次提交审核，代码审核者 对应bug记录进行回验。   
4.3、 代码讲解完毕后，代码审核者 给自己安排几个小时再对代码审核一遍。   
代码需要一行一行静下心看。同时代码又要全面的看，以确保代码整体上设计优良。   
4.4、 代码审核者 根据审核的结果编写代码“审核结果”并将“审核记录”和“审核结果”提交至GIT，审核记录 和 审核结果 参见附录I 审核记录、附录II 审核结果。   
4.5、代码编写者 GIT pull并根据“审核结果”给出的修改意见，修改好代码，有不清楚的地方可积极向 代码审核者 提出。   
4.6、代码编写者 bug fix等全部修改完成后提交 代码审核者 再次进行审核，通过审核则 代码审核者 更新本次审查结果并提交至GIT，审核通过对代码不能再进行修改，任何已通过审核的代码修改必须重新进行审核流程。   
4.7、代码审核者 把Code Review中发现的有价值的问题更新到"代码规范"的文档中，对于特别值得提醒的问题可群发email或QQ给所有技术人员。   
  
5. Code Review 标准   
代码审查的基础是我们的设计文档规范、代码规范、日志规范、测试代码规范，针对新增的业务场景和设计尚未有规范对应时应先确立规范然后再进行代码审核流程。