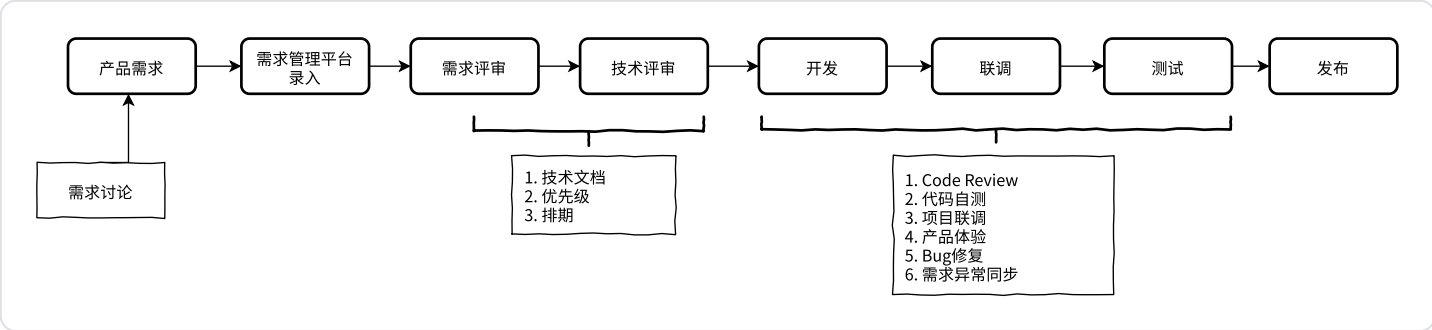


Code Review规范

目标

- 提高代码质量、提升代码组织能力，提前发现问题，降低修改/弥补缺陷的成本；
- 知识共享：促进团队分享，提高团队整体水平和一致性要求，帮助更多的人理解系统和产品功能；
- 可控性手段：团队成员快速了解和熟悉作者意图和想法，形成共建共享，提升可控性；
- 最优解方案：鼓励相互学习对方的长处和优点，弥补不足，逐步形成最优解方案；

流程和规范



技术方案编写

为了确保服务的稳定及方案的公开化，代码可持续性迭代，部分需求引入技术方案评审

- 开发排期在4天及以上的功能需求
- 代码的修改变动可能涉及主要服务或功能被停止或异常
-

Checklist

Code Review主要检查代码中是否存在以下方面问题：

代码的一致性、编码风格、代码安全、代码冗余、复用性是否正确设计以满足需求（功能、性能）等

- 完整性检查
 - 代码实现了设计文档中提出的功能需求
 - 代码实现了完整的数据链路处理，包括数据初始化、业务处理和异常处理

- 代码中不存在没有定义或没有引用到的变量、常数或数据类型
- **一致性检查**
 - 代码的逻辑是否符合设计文档
 - 代码中使用的命名、文件名、格式、符号、结构等风格是否保持一致
- **正确性检查**
 - 代码符合制定的标准
 - 所有的变量都被正确定义和使用 (ESlint)
 - 注释是准确的
 - 程序调用使用了正确的参数类型和个数 (ESLint::interface)
- **可修改性检查**
 - 代码常量是否易于修改(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
 - 代码只有一个出口和一个入口（严重的异常处理除外）
- **可预测性检查**
 - 代码所用的开发语言是否具有定义良好的语法和语义（杜绝奇葩语法: 例如string+变为number）
 - 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
 - 代码是否是否避免了无穷递归
- **健壮性检查**
 - 异常处理和清理（释放）资源
 - 代码是否采取措施避免运行时错误（如子对象检查、空对象等）
- **结构性检查**
 - 程序的每个功能是否都作为一个最简化的代码块存在（唯一性）
 - 循环是否只有一个入口
- **可追溯性检查**
 - 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
- **可理解性检查**

- 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
- **可验证性检查**
 - 代码中的实现技术是否便于测试
- **可重用性**
 - DRY（Do not Repeat Yourself）原则：**同一代码不应该重复两次以上**
 - 考虑可重用的服务，功能和组件（widget、component）
 - 考虑通用函数（Function）和类（Class）
- **可扩展性**
 - 轻松添加功能，对现有代码进行最小的更改。一个组件可以被更好的组件替换
- **安全性**
 - 进行身份验证，授权，输入数据验证，避免诸如SQL注入和跨站脚本（XSS）等安全威胁，加密敏感数据（密码，信用卡信息等）
- **性能**
 - 使用合适的数据类型，例如StringBuilder，通用集合类
 - 懒加载，异步和并行处理
 - 缓存和会话/应用程序数据

代码检查包括不局限于上述清单，提交人应在本地自我完成代码格式、架构设计、面向对象分析与设计等检查。

Code Review的步骤

1. 代码编写者和代码审核者坐在一起，由代码编写者按照UC依次讲解自己负责的代码和相关逻辑，从Web层->DAO层；
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；对这些bug记录在案。
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来。同时代码又要全面的看，以确保代码整体上设计优良。

4. 代码审核者根据审核的结果编写“代码审核报告”，“审核报告”中记录发现的问题及修改建议，然后把“审核报告”发送给相关人员。
5. 代码编写者根据“代码审核报告”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。
6. 代码编写者 bug fix 完毕之后给出反馈。
7. 代码审核者把Code Review中发现的有价值的问题更新到"代码审核规范"的文档中，对于特别值得提醒的问题可群发email给所有技术人员。