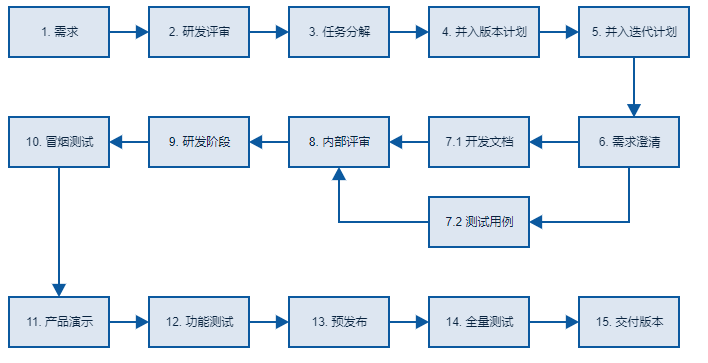
# 1. 整体流程



# 2. 详细描述

## 2.1. 需求

这里的需求是一个已经具象化的描述。其中包括以下：

产品原型：用来描述该需求应该是什么样的

思维导图：用来描述该需求在各种变量下功能应该如何展示

性能指标：给出最大可容忍值

UI/UX初步设计：大致定下开发时需要注意的地方，如需要考虑动画，图片透明度等等

UI/UX完整设计：需要在研发阶段前给出

## 2.2. 研发评审

前置条件：

* 需求文档（**必需**）
* 交互设计（**如果用到，必需**）

开发、测试、UI/UX及产品一起对这个需求进行评审，评审的重点是可不可以实现、实现的难易程序及对现有代码的影响。

对于原型比较复杂的原型参考 [需求设计流程](http://doc.dev.ft/pages/viewpage.action?pageId=2001108)

## 2.3. 任务分解

由系统工程师（SE）进行分解，如果 SE 对相应的具体模块不甚了解，可以与相应的开发一起将需求拆解成各个任务，拆解需求要注意，确保每个任务做到可单独交付给测试，

如果对测试可见，使用 Story 来标记，如果不可见（比如架构调整等），使用 Task 来标记。在分解时同时需要对该拆解后的任务评估工作量，任务的工作量不能大于 5人天，如果大于则需要将任务继续拆解。

## 2.4. 并入版本计划

由项目负责人（PO）在规划版本时将相应的**需求**并入进去。这里秉持一个原则，一个完整的需求必须规划到一个版本里，

除非这个需求被拆分成基本功能、动画效果等这种可以单独交付给用户的任务。

## 2.5. 并入迭代计划

项目负责人（PO）在规划迭代时将相应的**任务**并入进去。排迭代计划时要注意任务之间的依赖关系，确保每个人员在迭代周期内有事可做。

## 2.6. 需求澄清

由系统工程师（SE）在即将做这个功能时向具体开发人员及测试人员讲解该任务详情及其需求解决的问题

## 2.7. 开发文档

由开发人员在编码之前用专业的术语来描述当前的需求，要求覆盖大致实现原理，及相关各流程业务的代码处理情况。在编写该文档同时要考虑与已有功能的兼容及对未来功能的考虑，

避免只为当前功能而去写代码。设计要有一定的尺度，通过向产品了解业务的发展方向来给设计加入边边框框，使其不会沦为过度设计。

## 2.8. 测试用例

由测试人员与测试用例的方式来描述对需求的理解，该处流程不需要详细测试用例，只要保证功能用例即 0 级测试用例即可。

## 2.9. 内部评审

前置条件：

* 初期开发设计文档（**必需**）
* 0级功能测试用例（**必需**）

由需求负责人（产品）来拉起评审会议。评审内容如下：

1. 开发设计文档是否可满足当前需求
2. 开发设计文档是否可以适应将来的业务发展，不会导致大面积重构
3. 开发设计文档是否考虑相应的性能要求
4. 测试用例是否可以把当前需求的基本功能覆盖全
5. 测试用例是否可以把开发设计文档覆盖到

## 2.10. 研发阶段

前置条件：

* UI/UX 设计定稿（**如果用到，必需**）

说明：

由开发人员根据之前的设计文档进行开发。设计时以全集的思维去考虑，实现时可以只实现其中一个子集，留出对后续功能的扩展。在开发过程中以 TDD （测试驱动开发）的方式来进行开发，

编写单元测试用例，对实现进行检验，保证代码在编写上不出问题，编写代码时参考测试给出的功能用例。

## 2.11. 冒烟测试

由开发人员在功能完成后在内部包中进行测试，按照功能测试用例进行一一验证，重点保证功能的可用性，性能问题可以在设计值的 150% 以内即可。

测试时最好按照一个正常用户操作的流程进行，以解决正常用户问题为目标，对功能进行测试。

## 2.12. 产品演示

在冒烟测试通过后，开发人员认为当前功能已开发完毕，但其功能是不是产品需要的呢，这里我们需要给产品进行演示，在得到产品的认可后通过。

## 2.13. 功能测试

前置条件：

* 开发提供修改后的开发设计文档（**必需**）
* 开发提供冒烟测试结果（**必需**）
* 开发提供单元测试覆盖率（**必需**）

说明：

由测试人员在开发包上进行功能测试，重点测试当前包中所包含的需求，这里以基本功能通过为主，如有问题在不影响大局的情况以问题单的方式提出。

## 2.14. 预发布

前置条件：

* PO 提供版本功能说明（**必需**）
* PO 提供功能测试结果（**必需**）
* PO 提供安全风险报告（**待完善，可暂不提供**）

在解决完功能测试中的问题后，由 PO 发起预发布，此时代码应该进入稳定分支或封存（只允许当前版本的问题修复代码合入）。

## 2.15. 全量测试

此处不仅仅局限于测试人员，对于新产品来讲，可以鼓励更多的人去使用，从而进行用户场景全覆盖的测试，只要觉得不合理或做得不好都可以提出，PO 对提出的问题

进行甄别，如不影响用户操作的主场景的情况下，一些问题可以按后续优化来对待。

## 2.16. 交付版本

前置条件：

* 开发设计文档（**必需**）
* 测试用例（**必需**）
* 测试结果报告（**必需**）