

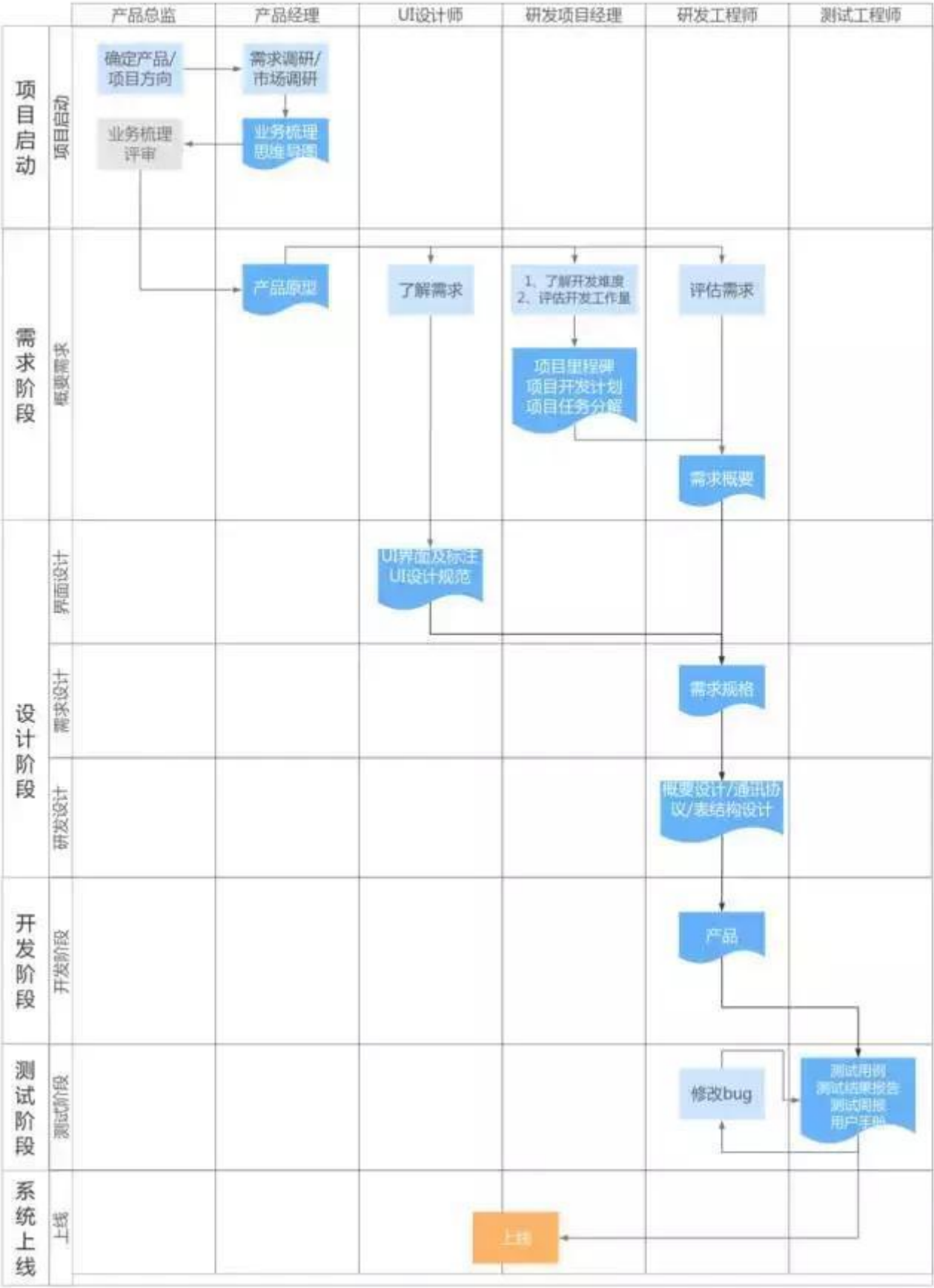
# 初见软件工程-从软件开发流程说起

---

突然说点什么也不知道从哪说起, 软件这个东西,说大不大, 说小也不小, 往大了说,基本上什么都离不开软件了, 往小了说,可能就是一个让人偷懒的工具吧 想整体的介绍的话, 很多很杂,包含了各个方向和部分(主要是这个部分.. 你啥也不知道)

## 软件开发流程是啥

借一张比较典型的一个软件开发到上线的流程图



项目启动阶段(解决了要做什么的问题)

这个阶段一般是确认项目的产品方向,营业的项目的话还要研讨是否有营业的可能, 个人的项目的话可能更多的是个人兴趣啥的

## 需求阶段(解决了有什么功能的问题)

这个部分一般是根据产品原型来确认是否有可行性的问题, 这个阶段产品的功能基本上就都有了,只不过还没实现

## 设计阶段(解决了产品是什么样子的的问题)

一般是ui出设计,开发也会参与设计是否有可行性, 基本上可能看到产品的样子

## 开发阶段(喜闻乐见的写代码阶段)

就是写写代码,实现下功能

## 测试阶段(保证软件的稳定可用)

测试这个部分,想要说也是一堆,有兴趣的话后面我可以多说点

## 上线(发布应用)

类似咱们现在的上架应用啥的

上面基本上就是一个比较完整的开发流程, 不过并不完整

## 说说写代码吧

我记得我刚刚开始系统的学习的时候, 一个学长和我们说, **真正写代码的时间其实只有整个软件开发周期的20%左右**, 最开始有的不以为意, 感觉再少也不至于这个样子, 不过后来一点点的发现, 20%已经不少了

下面说几个你可能感兴趣的问题或许方面吧

- 代码里都有什么 其实代码里面很简单, 不论是什么功能, 不论是什么样子的代码, 其实只包含了 **数据, 循环, 选择**, 三个部分, 任何的任何程序实际上都是这三个元素的组合.
- 怎么写代码 这个包含了两个部分, 一个部分是代码之内的部分, 这个是你看我们写代码能直观看到的部分, 及时那些代码, 用来自动执行处理事务的代码, 包含了各种的语言 比如python, java, c什么的不过一样通, 样样通, 学会一种语言后再看其它的语言有很多共通的部分, 只是熟练与否与经验的问题了. 另一部分的是代码之外的部分, 我一直认为代码之外的部分的才是整个编程中最重要的部分. 这个部分就是如何去写代码, 或者说如何解决问题. 之前让你看的那个文章属于这个部分, 是一种算法, (算法可以说是某些问题的优化处理方法). 代码这个东西, 写的越多就越认为再编程或者说软件体系中, 越没有代码的部分越重要. 代码只是一个实现的方法 怎么去做和怎么想到怎么去做才是最重要的. 说多了, 可能你不太理解, 或许写点小东西后会能理解点
- 程序员的主要方向
  1. 架构师 主要负责这个项目的架构和实现, 用来统筹整个项目之间的实现相关, 或者说搭建或设计整个项目的框架
  2. 前端 包括了android, ios, 网页 前端的主要特点是重点是数据的展示, 面向用户
  3. 后端 特点是对于数据的处理, 面向前端, 负责提供前端的数据支持和处理
  4. db(数据库) 这个和后端容易重叠, 不过也有专门的, 主要是数据的维护
  5. 运维 这个一个负责保证前端后端的稳定运营和硬件环境支持

6. 测试 对系统的各个部分进行测试 提前发现系统的问题 主要就是这几个部分, 其它有点零碎的其实也可以归为这几个部分

- 有例子不, 平时工作都是啥样的 有, 简单说说一个新的需求怎么实现的吧
  1. 确认需求, 可能是客户要求, 或者需要解决bug(这个部分是已经定位了bug): 这个部分一般是确认需要做什么, 需要达到什么样的目的(效果)
  2. 沟通问题, 这个比较宽泛, 因为实际的场景也比较复杂, 可能有很多的问题. 因为可能需要设计或者提供新的图片资源或者后台提供新的数据接口
  3. 解决问题, 一般就是写代码了 修改新的逻辑和效果啥的
  4. 测试 这个一般是自己测试啥的 有问题再改
  5. 发布 客户的的话一般是提供新的包 服务端的更新后台啥的
- 一个程序怎么运作的--(从客户打开应用开始)(这个部分比较杂, 有问题可以标记下再让我解释啥的) 比如说, 一个用户打开了咱们的应用, 最先接触的是客户端, 客户端一般要做的事情是和服务端通信, 交互数据 (这个数据是需要联网的, 不然怎么情况服务端, 现在服务端是后台也有打包出来的, 不过是在服务器运行的程序, 将打包好的包给运维, 运维在服务器中部署运行) 服务一般包含 逻辑控制部分, 就是客户端提供某个数据, 服务端进行处理, 除此之外还有数据库, 用于存储用户数据, 部分还有资源服务器(比如说用户图片之类资源) 其实也没啥说, 就是客户的请求, 服务增删改查返回数据什么的
- 啥是脚本(这个我熟) 脚本可以理解为一个很小的程序, 用来批处理文件, 实际上定义是 **脚本是不需要编译, 在运行时通过解释执行的程序**, 那么, 啥是编译(我太难了) 编译是讲代码转化为机器可以理解的机器码的过程 啥是机器码, 机器码(机器语言) 实际上就是机器执行的代码(二进制), 是cpu可以直接运行(不准确, 应该说是cpu控制器可以直接运行的部分) 除了机器语言还有汇编语言(一个偏向于人可以理解的, 类似 mov 4 这样的, 具有一定的可读性) 高级语言, 现在用的java, python数据高级语言 脚本就是不用打包, 运行时变成机器语言的小程序
- 脚本, 程序, 软件关系 软件= 程序+文档 程序 = 一堆脚本 + 打包(广义上)
- 怎么写程序.... 这个是另一个部分, 需要结合具体需求项目来了