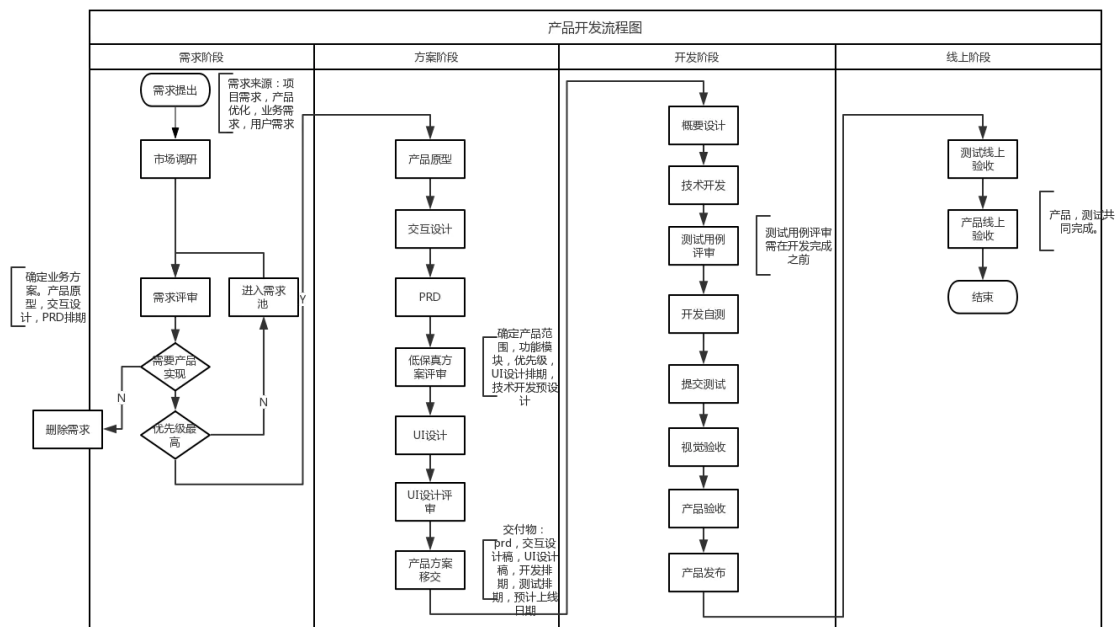
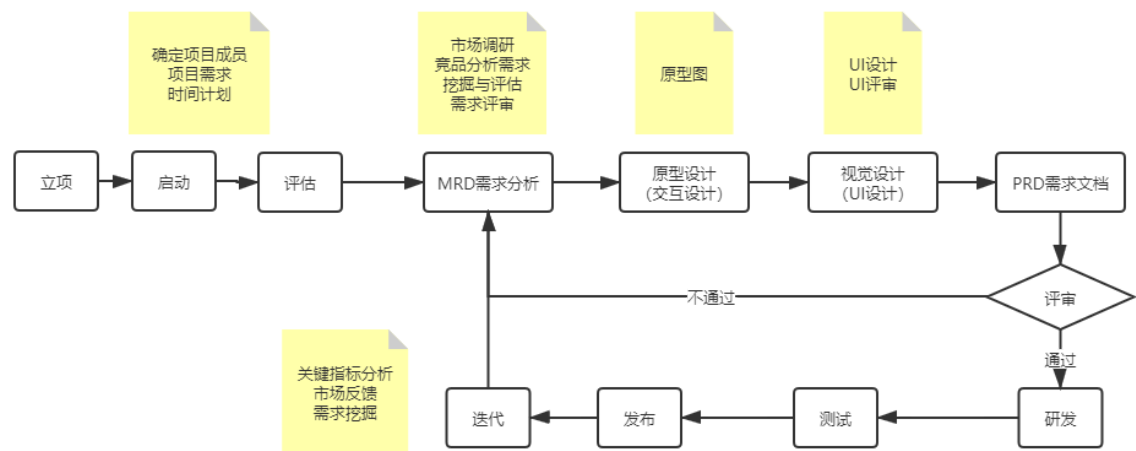


互联网产品开发流程



项目阶段	主要负责人	产物名称	产物描述
项目启动	项目经理、产品总监	项目组成员表	了解项目基本人员信息，包括角色、部门、职责、联系方式等
		项目范围说明书	了解项目背景与目的、项目目标、项目里程碑计划、评价标准等
		沟通计划表	了解项目利益干系人、所需信息、沟通方法与频率等
需求分析	产品经理	竞品分析报告	
		市场（用户）需求文档	市场分析、用户分析、竞品分析，梳理清用户的需求
		需求池管理	对需求的来源、需求的类型、需求描述、提交人、需求状态、需求优先级等进行管理
产品设计	产品经理、UI设计师	原型图	方便产品同事、UI同事，开发同事、测试同事沟通的产物
		产品需求文档	对产品概述、产品架构及业务实现流程、页面模块进行呈现
		UI图与标注图	
开发阶段	技术经理、开发工程师		
		代码	对代码进行管理，包括版本迭代管理
		接口文档	
		数据库表信息	
		项目注意事项	项目环境地址说明、项目注意点等进行管理
		对外接口管理	有些接口会统一封装给第三方进行调用，需要管理参数包括：接口接收方、接收方式、接受地址、对接版本信息等
测试上线	测试工程师	开发任务分配表	对任务名称、包含活动、对应处理人等信息进行管理
		周报	了解项目进度
		测试用例	检验软件是否满足客户需求
		缺陷统计与报告	对结果进行分析，发现产品中问题的本质，对产品质量进行准确的评估
		用户手册	用户操作说明书
		测试总结报告	
版本迭代	产品经理、技术经理	运营分析数据	支持版本迭代的运营分析数据
		需求版本管理	
		原型图版本管理	
通用		评审会议纪要	包括市场需求评审、原因与UI评审、技术评审、上线验收评审

主要分为五个阶段：项目启动，需求阶段，产品设计，开发上线，版本迭代

立项 启动 评估

立项：确定要做一个什么产品

启动：确定项目相关人员，项目需求，产品原则，时间排期

评估：评估产品机会

需求阶段

需求收集

来源：老板提出，项目需求，产品优化，业务需求，用户需求，运营需求，增长需求，商业化需求等

方式：头脑风暴、用户调研、用户反馈、竞品分析和数据分析

头脑风暴

产品、运营、视觉、技术都可以参与。人数不要过多，超过5个效果反而不好。围绕着一个核心问题，自由发挥发表观点，不评论对错。产品要做好会议控制，避免漫无边际的讨论，同时做一个好的timer。如果氛围尴尬，可以尝试引导做问题拆解，将一个点，解析成几个关键点讨论。或抛出设想和问题。

注：一定要有人做记录，可考虑录音

用户调研

方式：问卷调查、用户访谈、可用性测试

注意：多聆听、观察和思考

注：可用性测试需要专业的观察室，往往只有大厂会做，也往往是交由用户研究员来完成。一般是已经开发完成某项功能，让一群有代表性的用户对产品进行操作。观察员在旁边或者监控设备中观察和记录。目的是发现用户真实使用过程中，会遇到的问题，用以提升产品可用性。

用户反馈

用户反馈可以跟用户增长结合在一起做。

定个策略，比如使用多少时长后，或者完成某项任务后，弹出一个邀请评分窗。用户要是选择四星或以下，点OK键，则进入反馈流程。如果是五星则引导去应用商店评分。

这种做法巧妙在于，不仅可以很有效的过滤掉负面评价，让负面评价留在反馈系统，而且还能让应用商店的好评获得大幅增长。

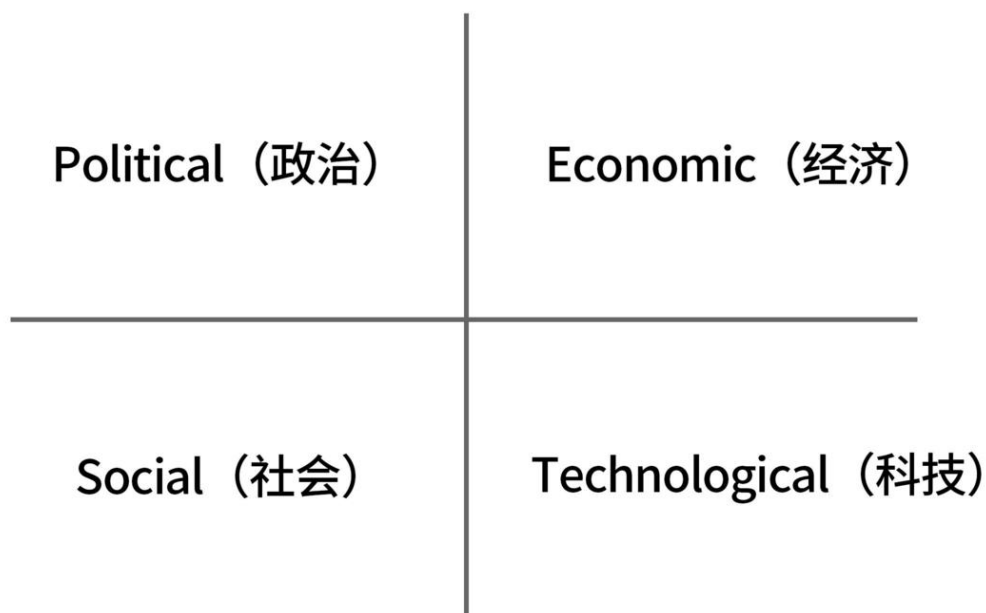
市场分析

考察大环境的情况

分析法——PEST和SWOT

PEST

PEST 是一种宏观环境分析模型，所谓PEST，即 Political（政治），Economic（经济），Social（社会）and Technological（科技），从四个维度分析大环境。



SWOT

SWOT即S（strengths）是优势、W（weaknesses）是劣势，O（opportunities）是机会、T（threats）是威胁。清晰地把握全局，分析自己在资源方面的优势与劣势，把握环境提供的机会，防范可能存在的风险与威胁。这是SWOT分析法的目标所在。



竞品分析

前期可以做比较全面的竞品对标，横向的纵向的跟竞品进行对比。

对比产品之间的优劣势，比如资源、渠道、技术等。对比目标战略方向、用户选择、产品策略。对比产品功能点。对比竞品是如何做用户体验，怎么处理逻辑、界面层级和细节的。等等。

如果是一个新来者，还可以查看对手的迭代记录，从而绘制出对手的迭代图谱。

竞品分析其实很重要的在于竞品追踪。时常留意竞品做了哪些迭代。你要对这些迭代做思考，思考迭代背后的动机是怎样的，可能会产生哪些效果，这些迭代功能是否要跟进。

除了通过看对方用户反馈和情报了解外，有时候确实还是不好判断效果如何，因为数据毕竟是掌握在对方。这时候可以看看对方接下来做出的调整和修改，用来佐证判断。当然这种方式周期很长，你也可以尝试做AB测试。

数据分析

要从根源解决问题才行。**所以拆解问题很重要**，找到问题存在的所有可能性。再寻求统一的解决方案。

对问题进行拆解或者归因，再反向寻求解决方案。这时候就需要从业务逻辑出发，脑海中去模拟使用场景。理清脉络，顺藤摸瓜。再结合数据看，就能比较容易发现问题。

除自身数据外，还可以参考一些数据调研机构的分析报告，比如艾瑞资讯等对互联网行业做的数据分析。还可以试试百度指数，淘宝指数等工具。

海外市场的分析工具：

- Appsflyer /Adjust:用户获取渠道分析归因分类平台,方便统计不同渠道给同一谷歌商店上架产品的贡献数量和转化效率。
- Appannie :不同国家APP市场的行业数据分析、竞品数据分析等。
- Google analytics :产品内部数据检测分析、渠道分析等

盈利分析

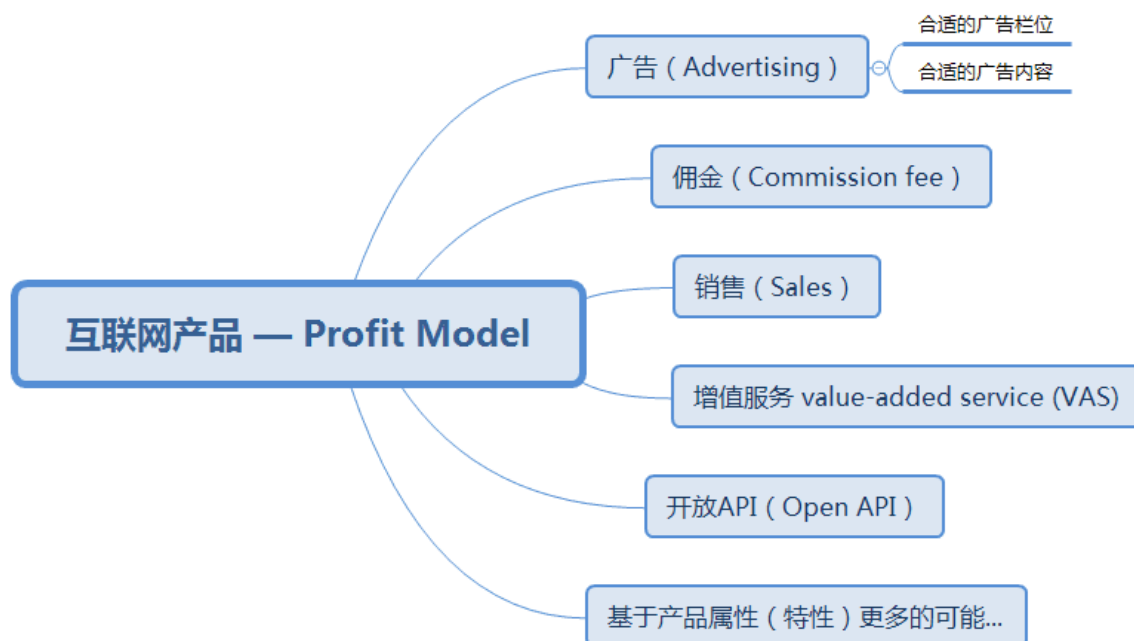
盈利模式

指按照利益相关者划分的企业的收入结构、成本结构以及相应的目标利润。

通俗来说，盈利模式就是企业赚钱的渠道。有效的盈利模式分析有利于企业从既有的单一赚取利润的思维定式中解脱出来，寻求更为多元的利润来源方案。那么互联网产品在进行盈利模式设计时首要考虑的是什么呢？是用户。互联网产品盈利模式设计的首要原则就是“以用户为导向，以利润为中心”。而定义一个好产品的基本要求就是“满足用户的需求，完成企业的指标”。

互联网产品盈利模式设计应以用户价值为核心，并需要在市场细分和客户偏好分析的基础上，通过有效的定位来抓住高利润的客户群，实现稳定盈利。

互联网产品盈利模式思维导图



需求分析&筛选



筛掉明显不合理的需求

这个阶段的判断标准就是需求的合理性，用你的经验、专业知识，甚至是直觉，过滤掉大部分需求。比如，当前技术不可能实现的或意义不大的、投入产出比低的、明显不合理的需求。

因为从各种不同渠道会收集到大量的需求，为了提高效率，你不得不这么做。当用户量达到一定规模后，光每天收到的用户反馈就够你喝一壶的。甚至因此，你不得不做一个用户帮助和反馈系统，来帮助你过滤和初步加工需求。

有个简单的判断方法：这需求做了会怎样？不做又会怎样？

如果做不做没多大区别，甚至做了会起到负效果，那就不需要犹豫了，可以直接过滤掉这条需求。

挖掘用户的潜在需求和动机

这是用户需求进化为产品需求的关键一步。

用户需求：need，用户想要的。 产品需求：solution，解决方案。可以是推荐算法优化、界面布局调整、新功能点，甚至是新产品等等

以上是用户需求和产品需求的定义，用户需求是用户想要的东西，产品需求是满足用户需求的解决方案。用户毕竟不是专业人士，考虑问题的出发点是用户自身，这就需求产品经理去挖掘用户的潜在需求和动机。

最著名的例子就是：If I had to ask customers what they want, they will tell me: a faster horse. “如果我当年去问顾客他们想要什么，他们肯定会告诉我：一匹更快的马。”

如果福特先生不去挖掘用户背后动机的话，可能就去研究马的配种问题了，而不会有福特汽车了。

那么如何挖掘用户潜在需求和动机？

用户需求的产生，带有很强的不确定性，受环境、情绪等各种因素影响。所以光看表面描述不行，得有代入感的去感受。可以通过以下几个要素进行场景分析：

用户需求=谁（用户特征）+在什么情况下+想满足什么？

比如你是Todo list产品经理，调研过程中，某用户反馈添加过程太麻烦了，没法批量添加。追问为什么需要批量添加后，对方回答，她是一名学生，想要把课程表导入到list中，方便查看。这时候你就明白了，对方是一名大学生，在添加操作的环节遇到了麻烦，她是想要添加课程表。这时候你就可以马上找到解决方法：可以利用OCR技术，只需扫描一下，直接将整张课程表导入进来。再深入思考一下！添加课程表就是最终的目的吗？显然不是，对方添加课程表后，一是为了方便查看，二是为了上课时间快到时提醒她上课。那么就可在导入后，行程一个可视化的日程表，提前提醒上课。

这就是需求挖掘过程。

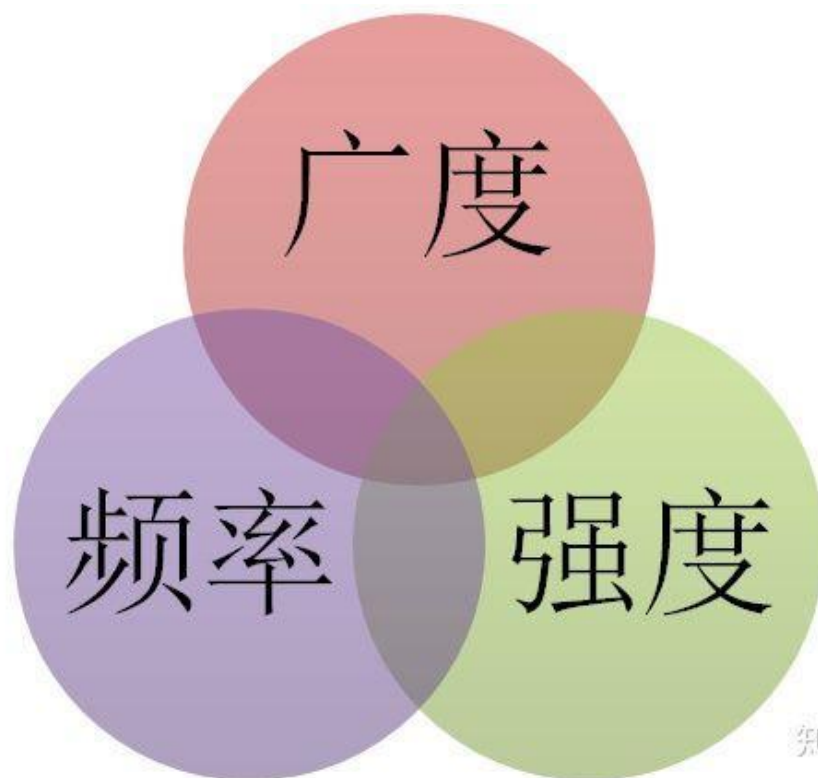
需求归类整理

挖掘到用户的真实目的后，会发现，多个看似不同的需求背后，原来是出于同一个目的。这就可以将这几个需求归纳为同一个。

接着可以通过如下维度归类整理需求：

1) 判断是否有价值：广度、频率、强度

如果使用人数很多、频率很高、需求也很强烈，那当然是好需求。如果这三者都不沾边，可以判断为无价值需求，可过滤掉。



需求优先级评审

为了让开发对需求的可行性、开发难易程度做大致评估，以及提供实现思路（不同的实现方式对逻辑流程和原型会有影响）。部分需求可能要分期开发。会议之后输出需求优先级、争议点和难点。开发在会后就可以开始技术预研。

需求分类方法

1. 投入产出比ROI：商业价值/用户价值-开发成本

要考虑需求的实现成本（人力、时间、资源、运营等因素）以及收益（商业价值/用户价值），综合考虑是否将其纳入本阶段的需求库，还是放到下一期实现。

2. 马斯洛&七宗罪需求分析法

马斯洛需求理论：生存需求、安全需求、社交需求、尊重需求、自我实现需求

七宗罪：傲慢，嫉妒，暴怒，懒惰，贪婪，饕餮，以及欲望

分别从需求的层级和人性的角度，对需求进行深层次的把握

3. 痛点分析法

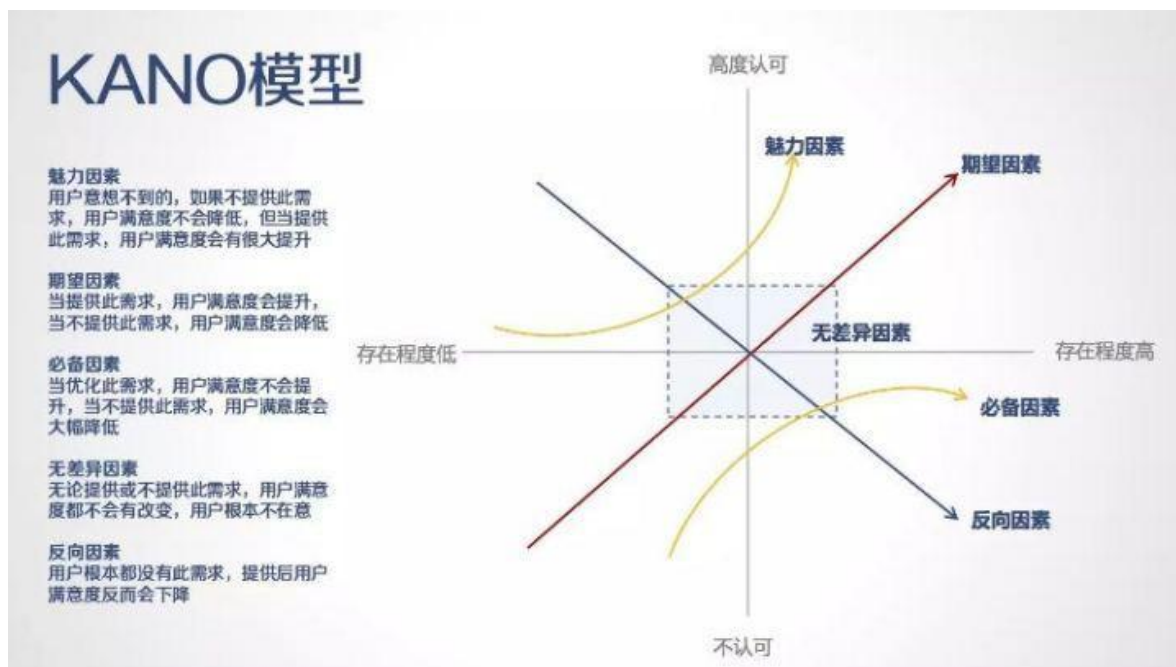
将需求分为：痛点、痒点和兴奋点需求。

4. KANO模型分析法

可以将KANO模型理解为是痛点分析法的加强版。以分析用户需求对用户满意的影响为基础，体现了需求实现程度和用户满意之间的非线性关系。

将需求划分成如下5类：

基本(必备)型需求--Must-be Quality/ Basic Quality 期望(意愿)型需求--One-dimensional Quality/ Performance Quality 兴奋(魅力)型需求-Attractive Quality/ Excitement Quality 无差异型需求--Indifferent Quality/Neutral Quality 反向(逆向)型需求--Reverse Quality



KANO模型

匹配产品定位、产品目标和资源情况等限定条件，做最后一轮筛选

需要考虑产品定位，符合战略目标、目标用户和既定的功能范围。做需求要为产品的核心功能服务，不然容易功能越做越多，庞大而冗余，反而导致用户流失。

确定目标用户很关键，确定目标用户，需要综合考虑受众群体大小、需求相关性、商业价值和已有资源。总得来说功能要为有价值的用户服务。

举例：

比如一款音乐产品，受众年龄15岁到65岁都有，但你不能说目标用户就是15岁到65岁的人群。因为不同年龄段的受众，甚至需求都是相悖的，兴奋点也不一样。这产品就不好设计了，最后搞不好两边用户都不讨好。通过分析发现，用户群体可以划分为休闲放松、时尚潮流、极客等类型。再综合考虑受众群体大小、需求相关性、商业价值和已有资源，你可能就选择了休闲放松这个群体规模大、频次高的人群作为你的目标用户了。当然这里是简单介绍，每款产品会有不同的选择。但总的来说，目标用户一定要明确，可画像的。

定义优先级

在此之前需要判断产品所处的生命周期，因为不同产品生命周期的侧重点是不一样的。

(1) 产品初期 (0-1)：做最小可行产品(Minimum Viable Product,简称MVP)，满足用户核心需求，快速上线，快速迭代。积累种子用户。此时产品不宜做得大而全，方便调整产品方向。如果是内容型产品，还需运营好社区基调，控制用户导入，比如知乎早期采用的邀请制。

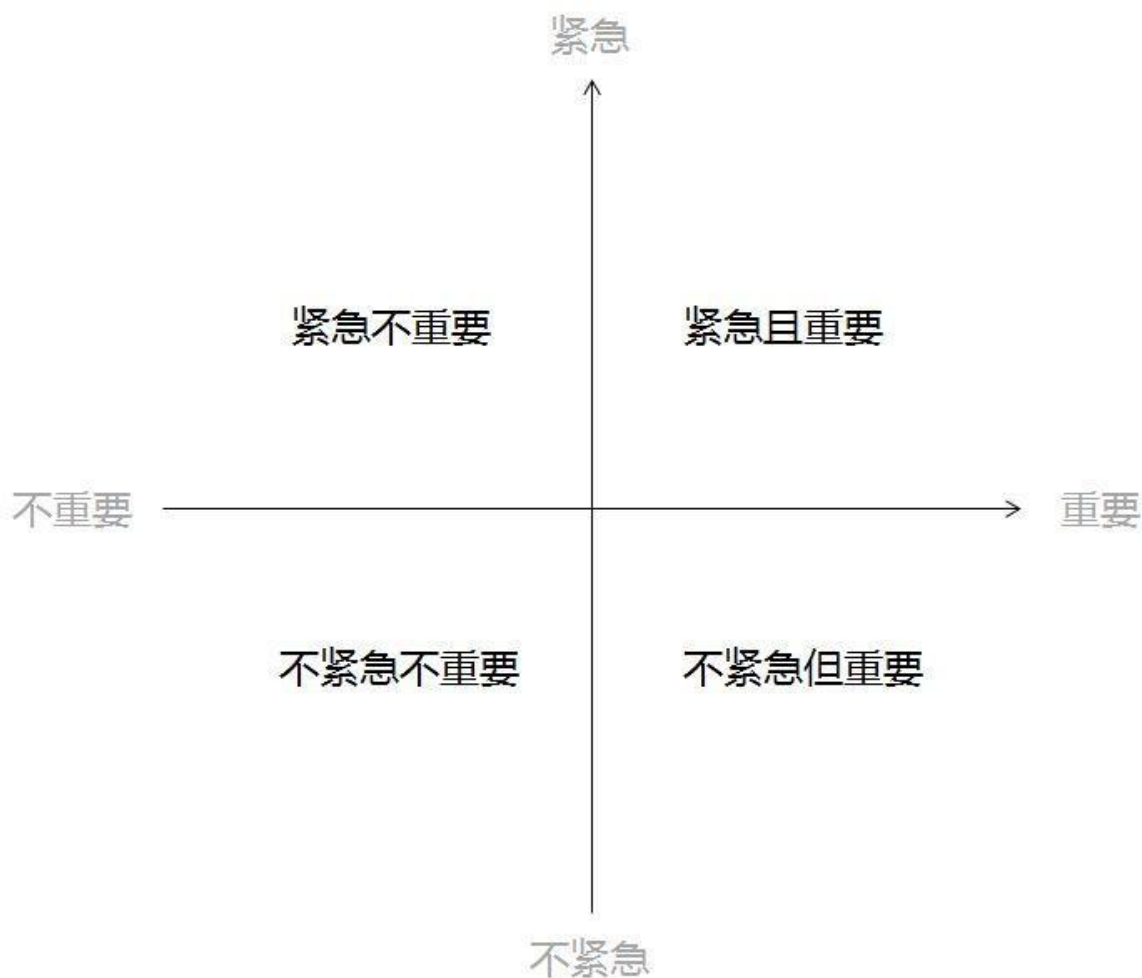
(2) 成长期：继续打磨核心需求，完善功能短板，让产品朝着指定方向发展。这时候会加大运营投入，用户大量导入，需求激增。这时候团队压力很大，要控制好需求，把握好核心用户，把资源用在刀刃上。同时重点关注留存和活跃，提高粘性和使用时长。

(3) 产品成熟期：不断打磨产品，巩固产品壁垒，制造兴奋性需求，挖掘潜在用户，扩大用户规模。同时要开始考虑变现。

(4) 产品衰退期：尽量延长产品生命周期，持续带给用户新鲜感，留住用户。扩充品类，孵化新产品。

总的来说，符合当前公司发展目标的优先级高，反之降低优先级。

接着就可以结合重要紧急四象限，排出优先级。这里就需要具体情况要具体分析了，没有统一评定标准。



重要紧急是很好的排优先级方法，有的公司会进行多轮的需求评审，优先级评审阶段，在小黑板上画出重要紧急四象限，将需求根据讨论写在白板上面。根据需求的分布情况，优先级就一目了然了。

当然别忘了，需求要做到什么层度，是简单做还是复杂做，是不是分阶段做。比如登录需求，是用三方登录还是需要自建一套账号体系。比如微信登录，熟悉的开发可能1人/1.5天就搞定了。自建账号体系就相对比较复杂了。

具体做到什么层度，就需要产品经理自己定义了。技术评审会议的时候，开发特别关注这个。

我下篇文章会介绍，怎样从需求到落地。

做产品的都知道要道和法要结合，法即是方法，如上文讲的都是方法。道就是方法论，在此做下方法论上的总结，刚好以此作为本篇文章的收尾。

本篇文章思维逻辑是这样的：当遇见一个问题，先明确问题和目标，再对问题做拆解，最后形成可执行性的方案。再接下来就简单了，就是执行、检查和修正。可以说，定义好问题后，问题就解决一大半了。

首先，对问题做明确的定义：

需求分析是：从获取到的用户需求中，通过一系列的判断方法，结合产品定位、产品目标和资源情况等限定条件，以提炼出产品需求。

接下来，对问题做拆解：

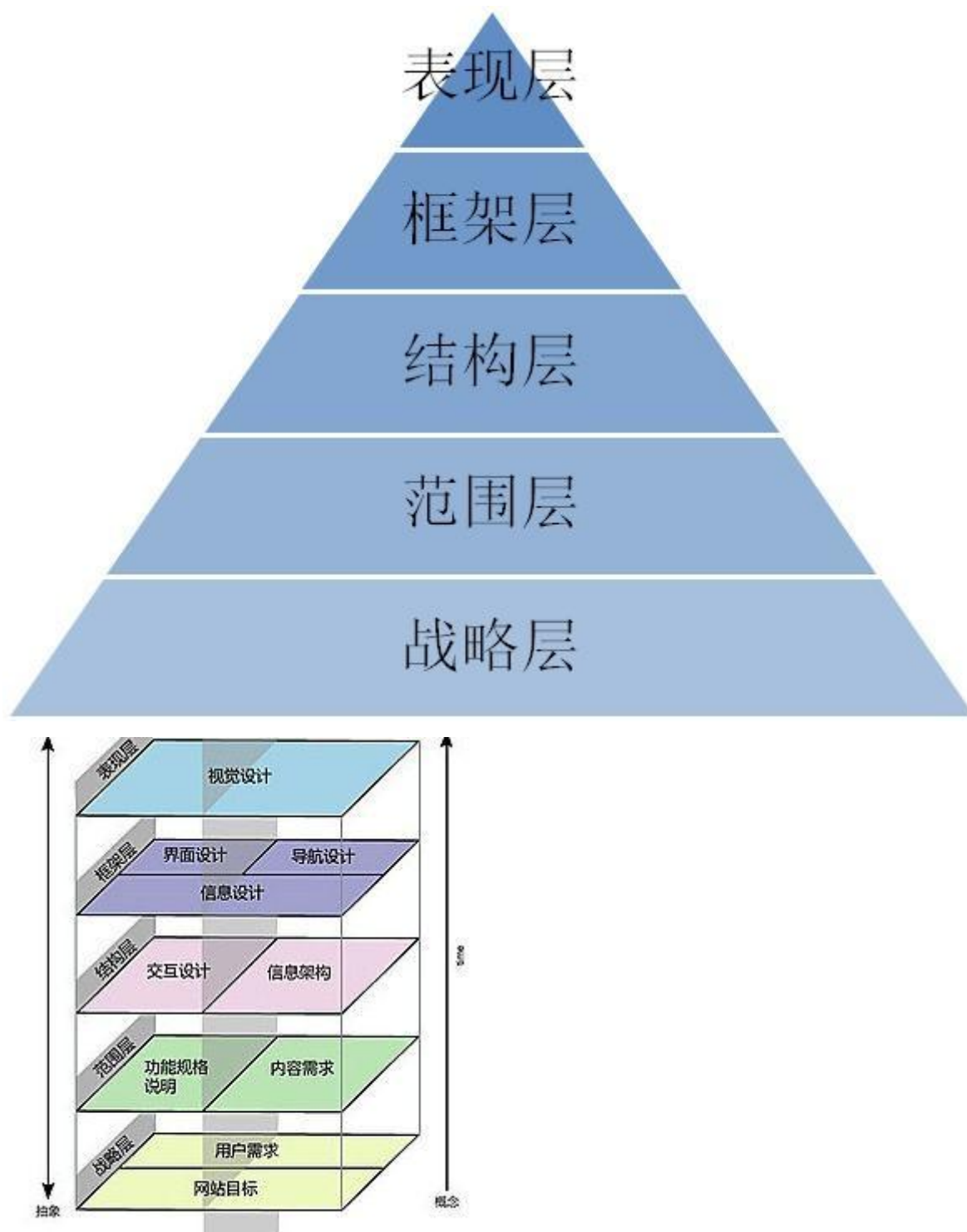
我们通过定义可以看出：

1. 这是从用户需求到产品需求的过程；
2. 需要一定流程和限定条件；
3. 目标是提炼出产品需求。

第三步，明确需求分析与筛选流程，形成可执行步骤：

1. 筛掉明显不合理的需求
2. 挖掘用户的潜在需求和动机
3. 需求归类整理
4. 匹配产品定位、产品目标和资源情况等限定条件，做最后一轮筛选
5. 定义优先级

产品设计



注：用户体验要素5层模型

战略层：产品目标及其目标用户（做什么、为谁而做？） **范围层：**功能及其内容需求整合（需要做哪些？） **结构层：**交互设计及其信息架构（怎样做？） **框架层：**界面设计、导航设计和信息设计（怎样做？） **表现层：**视觉设计（要做成什么样子？）

产品设计过程:

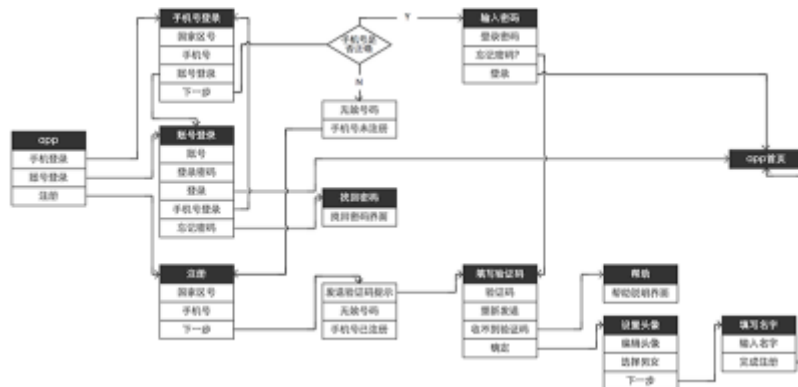
1. 将需求归纳整理成**用户任务**
2. 多个用户任务交织成**信息架构**
3. 将信息架构具象化为页面流程图，并通过**界面草图**的形式表达出来
4. 接着根据设计原理，让界面草图进化为**页面原型**
5. **最后加点调料**，通过情感化设计和游戏化设计，赋予产品魔力

流程图&设计草图

设计原型前，先进行结构层和框架层的设计。也就是流程图到设计草图（纸面原型）。这一步不能跳过，前期考虑周全，会节约大量的时间，避免后面修改起来牵一发而动全身。这个环节很多同学会去研究竞品或者寻找相似设计，这本身没什么错，但最好不要这样。容易固化思维。你要先把自己脑中想象的东西画出来，然后再去对标竞品，如果对方好的设计你可以借鉴，不好的地方那就你设计的独到之处了。

功能流程图

在有一些比较复杂的流程的时候，怕开发人员或是设计人员不是能理解的情况下，要把流程图画出来。但是一个完整的项目只用一个流程图是不可能完成的，所以需要按照模块的流程逐个进行分类，分类以后按照功能的不同、简易程度，分别画出流程。



解析业务流程图与产品流程图

用户行为/用例

产品的功能点、主要功能卖点

界面布局

文字文案、各功能描述、弹出框内容文案，各模块之间关系。

信息架构

各模块业务逻辑、结构框架图

原型图

规范的交互原型图结构由如下五要素组成:

- ## 1. 更新日志

2. 迭代记录
3. 产品结构图
4. 原型图
5. 交互说明

产品结构图



产品结构图会忽略掉很多细节，只展现页面层级结构，而且只表达正常情况下的页面关系。目的还是为了简单，一下子就看得懂，方便记忆。

比如，会忽略掉复杂的页面跳转关系，而只表达主干道上的页面流程。会忽略掉众多的状态判断，而只表达正常情况的页面流程。

原型图

好的原型工具有Axure RP、墨刀等等，学一款原型工具很简单，找个视频看一下，10分钟就会了。重点在于要把自己的想法通过原型表达出来。原型也是跟视觉、开发和测试沟通的桥梁。原型有低保真和高保真，甚至动态原型图之分。这个看团队需要。

UI设计&UI评审

视觉设计

交互设计一旦通过，就可以进入下一步的高保真图的设计，视觉设计组会根据产品经理的描述，设计出最新的视觉效果(高保真图)，一般采用ps等设计软件。在交互评审通过后，负责把控UI整体风格和所有视觉效果的设计，主要输出：VI选择的方案、所有视觉效果图、资源包等。

功能/操作界面

交互设计（规范）

需求评审会议

拉领导、开发、视觉、测试等多方参与。评审内容以原型图为主。第一次需求评审会议，需要给团队介绍产品目标、目标用户、设计思想等等这些问题，确保大家目标一致，思路统一。会后同样也要输出原型修改点和争议点等等这些内容。一般需求评审会议开个几轮都是正常的。有的公司要求在需求评审阶段就要输出PRD文档，有的公司不鼓励写PRD文档，毕竟写PRD文档也是一项耗时的工作，开发还不一定看。当然，这因公司而异。

PRD文档（Product Requirement Document）

PRD详细需求文档一般与视觉设计是同步进行，主要是细化MRD里的功能以及详细流程、文案等细节，主要还是产品经理负责。

PRD详细需求文档，一般包括功能流程图、产品的功能点(框架脑图)、主要功能卖点，模块的内容，文字文案、各功能描述、弹出框内容文案，各模块之间关系，以及各模块业务逻辑、结构框架图等。

涉及到负责的流程，需要完整的流程图，需要按照模块的流程逐个进行分类，分类以后按照功能的不同、简易程度，分别画出流程。

开发上线

开发排期

开发排期技术负责人会把控，但产品作为总协调人，需要统筹开发和视觉进度。开发和视觉同时启动，这没什么问题，但视觉进度一定要快于开发进度，这是肯定的。开发排期有粒度的概念，即开发会将需求拆分成小的功能点，每个功能点完成时间是以天为单位、0.5天为单位还是以小时为单位。这个时间单位就是粒度。这就需要产品去要求开发同事了，不然他到时排期给你个以周为单位，这就没有多大指导意义了。颗粒度越大，进度越不可控。因为开发过程中，有很多不可控因素，比如中途插入更高优先级需求，这时候你需要清晰的知道，在哪个环节延误了，已经延误了多长时间。以方便及时调整开发计划。一般大的功能点可以以天为单位，小功能点以0.5天或小时为单位（比如需要及时响应的运营需求）。

开发项目跟进

有点类似项目管理，协调资源、催进度（周启动会议、周总结会议）、关心开发实现上的问题、安排好后续测试和上线环节。这些都是分内的事。所以也不是人人都是产品经理啊，产品经理要求还是比较全面的。

开发进度管理

研发流程步骤：概要设计->详细设计->设计评审->编码阶段->测试阶段->运维上线

团队协作

研发人员步骤：前端->架构->后端->数据库->测试->运维

前端工程师

负责把视觉设计后的高保真图，转换为html、css，利用js实现其交互效果。大家熟知的前段工程师。

架构师

一般产品进入研发阶段，大的项目需要做技术的概要设计和详细设计，保证技术方案的稳定性、可扩展性、性能等多项指标，这里就会涉及到架构师的参与,确保技术方案的可行性。

后端工程师

实现后端功能开发，让功能可用、易用，「程序猿」是最直接的描述，写代码的牛人或者普通人。

测试工程师

测试工程师，会测试产品的缺陷，在各个终端的适应性，以及产品在高访问下的性能测试。

运维工程师

测试通过后上线，会涉及到线上硬件运维。

项目管理

协调技术、产品、运营等部门把产品更好的实现。

需求管理

视觉验收

视觉验收有视觉设计师负责，主要是看看视觉上有没有需要调整的地方、开发是不是把资源合入错误了、开发实现是否跟视觉标注一致等等。 产品验收当然是啥都管，除了检验开发是否按需求文档实现外，视觉、bug、逻辑实现问题、响应速度加载速度、边界问题（如弱网、断网、服务器失联、空白页）等等都要检查到。 产品验收也不是非得等到开发完成后，中途穿插几轮验收也未尝不可。

测试

测试同事在PRD文档确定后，就可以安排开始编写测试用例。一般团队也有测试指标，达到指标后，方可上线，比如无一二级bug，三级bug（不影响体验的小概率bug）少于几个，功耗、性能指标等等。当然，上线前，还可以安排可用性测试、A/B测试、压力测试等等。

发布上线

上线前准备好上线材料，跟运营同事配合做上线工作。上线后要实时监测各项数据及用户反馈，做好预警方案。 如果是重大功能上线或影响范围比较大，为了可控，避免造成突发事件，还可以安排一轮灰度测试。

流程：

1. 用户教育
2. 销售培训
3. 推广方案
4. 运营策略
5. 产品定价
6. 发布公告
7. 发布

版本迭代

1. 发现问题&需求收集
2. 用户反馈
3. 功能改进
4. 数据分析
5. 运营策略调整
6. 数据挖掘