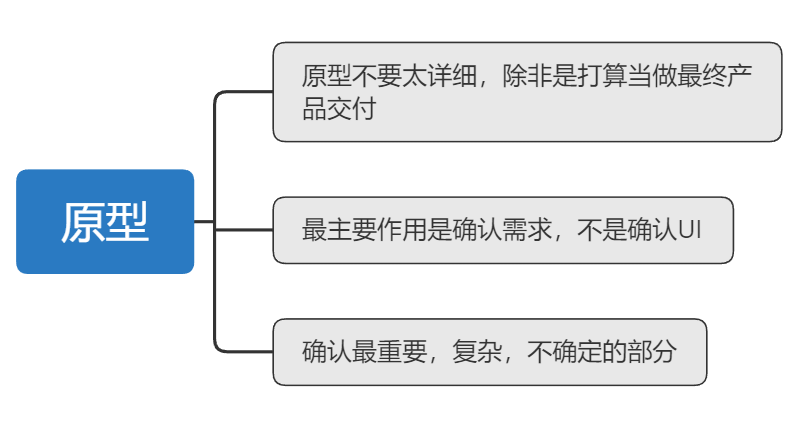
### 通过原型来减少风险

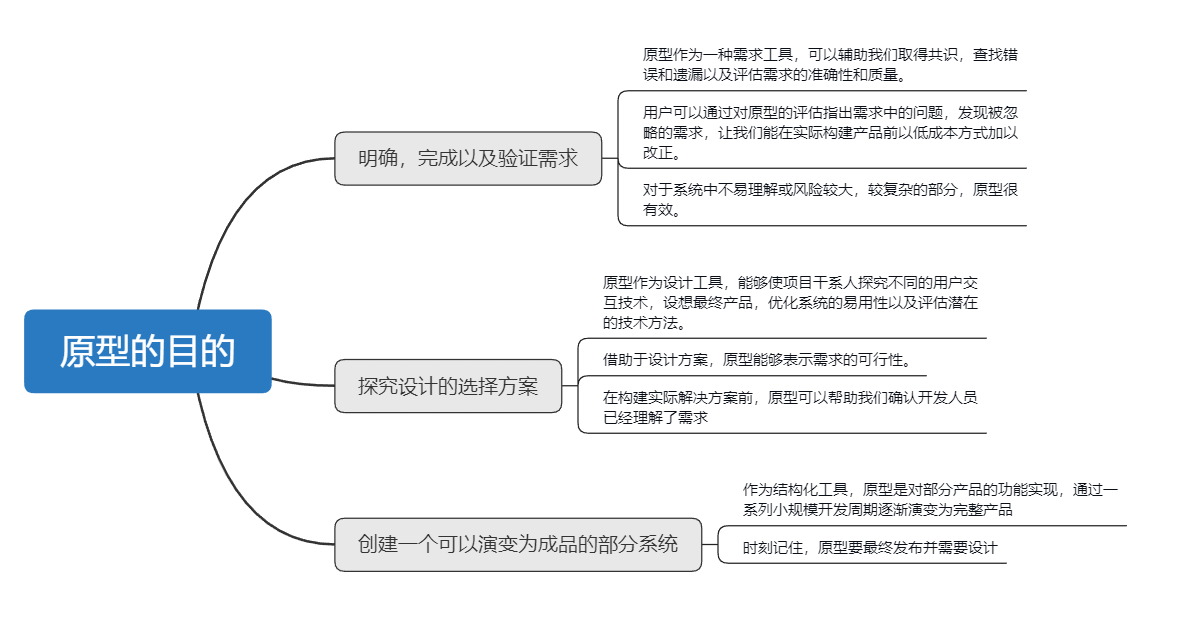
原型的主要作用是通过对新系统建模或者给用户提供一个**粗糙**的新系统，激发用户思考并引导出需求。

只要存在使客户，开发或双方都不确定或不清楚需求中的某些部分，原型的引入就可以很好地表述所有重要的客户关注点。



## 原型的定义及其动机

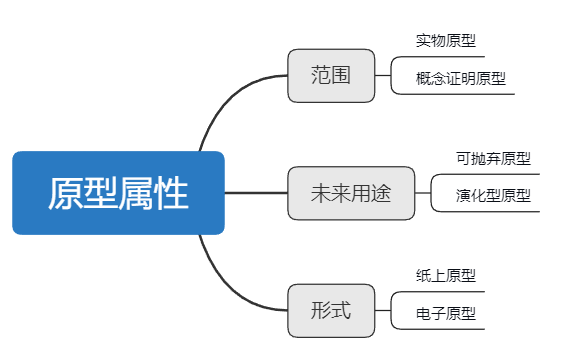
**原型能够实现的三个主要目的：**



创建原型的主要原因是想在开发过程中尽早解决不确定的问题。重点要放在高风险或已知不确定的功能上。这样能帮我们发现和解决有歧义，不完整的需求。

**原型属性：**

* 范围：**实物模型**这一类原型重点关注用户体验；**概念证明原型**探究的是提议方式方法的技术合理性。
* 未来用途：**一次性（可抛弃）原型**在产生反馈信息以后会被抛弃，**演进型原型**则通过一系列的迭代发展成为最终产品。
* 形式：**纸上原型**是画在纸上，白板上或者画图工具中的草图。**电子原型**只针对部分解决方案的可工作软件组成



## 实物模型和概念证明

实物模型通常被人称为**水平原型**，实物模型意味着它实际上没有实现行为。

它展示的是一些UI屏幕的一些表现形式以及其之间的导航，不包含或是很少包含实际的功能实现。

实物模型可以展现用户可用的功能选项，用户界面可能只看到一条消息，或者发现一些控件没有任何功能。

演示型模型看起来好像应该能够执行一些有用的工作，但其实不然。这样的模拟往往足够帮助用户判断是否有遗漏，错误或者不必要的功能。

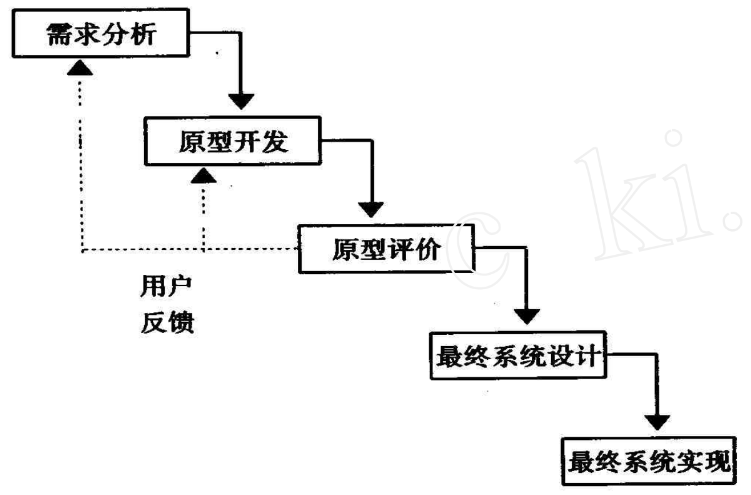
使用抛弃型演示型模型时，用户应该重点关注概要性需求和工作流问题，不要被屏幕元素的确切外观分散注意力。

概念证明又被称之为**“垂直模型”**，它在所有技术服务层面上从用户界面实现一部分应用功能。它触及了系统实现的所有层次，常用于测试预期架构方案合理性，算法优化，评估预期数据库模式，确认云解决方案的稳健性。

## 抛弃型原型和演化型原型

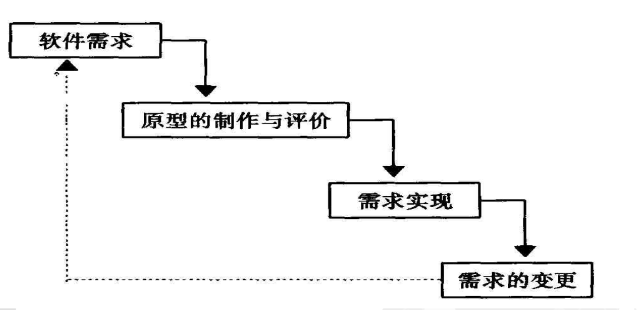
如果想解释一些问题，解决不确定性以及改进需求，就可以**快速，低成本**的创建一个**抛弃型原型**。创建抛弃性原型时，会忽略成品软件构建技术。相比健壮性，可靠性，性能和长期可维护性，可抛弃性原型更注重快速实现以及快速修改。

当团队觉得需求不确定，有歧义，不完整或者含糊时，就可以选择建立抛弃型原型。



相比较于抛弃型原型，演化型原型更加稳固。用前期的迭代来调整未来开发周期的方向，并通过一系列迭代来完成产品的构建。这就是演化型原型的实质。

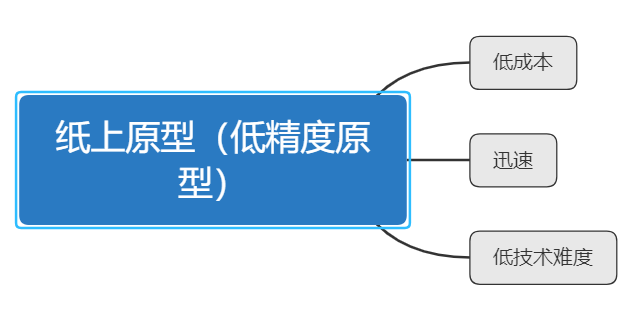
构建**演化型原型**时，一开始就要考虑**健壮性**，写产品级质量的代码。需要重视软件架构和稳健的设计原则。每一轮迭代都会在实现部分需求的时候获取下一轮迭代需要修改的内容。整个产品就是由一系列对原型进行周期化演化积累而获得的。



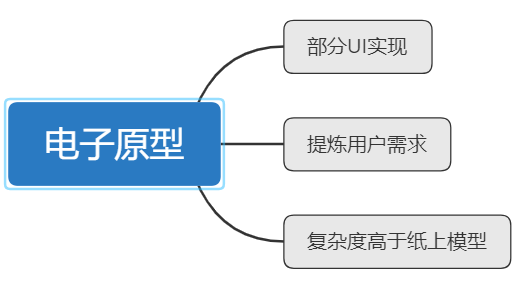
|  |  |  |
| --- | --- | --- |
|  | 抛弃型 | 演化型 |
| 演示型模型 | 澄清与提炼用户需求和功能  识别被遗漏的功能  研究UI方法 | 实现核心用户需求  基于优先级实现额外的用户需求  实现和优化网站、  使系统与快速变化的业务需要相适应 |
| 概念证明 | 演示技术的可行性  评估性能  获得更多知识以提升估算能力 | 实现和拓展核心多层级功能以及通信层  实现和优化核心算法  性能测试和调优 |

## 纸上原型和电子原型

纸上原型能帮助我们探究一个要实现系统的部分外观。



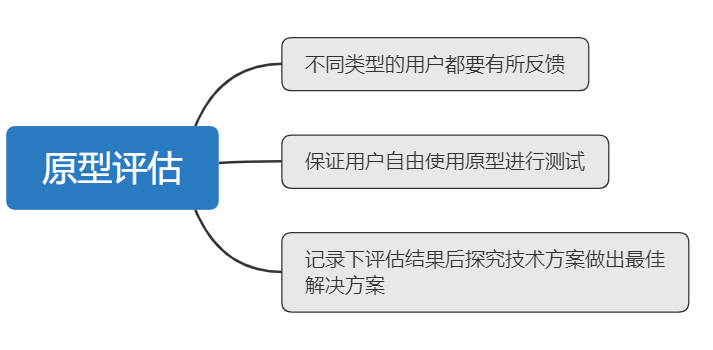
纸上原型不需要考虑控件出现的具体位置和具体样式。纸上原型可以促成快速迭代。



电子原型更加具体，复杂，但无论是电子原型还是纸上原型，都不应该涉及到过于详细的UI设计。我们在确信已经充分了解到必要功能需求之前，应该将原型的目的定位在提炼需求上，而不是视觉设计！（比如说字体再深一些，控件往上一点这种视觉设计在原型设计上不是主要目的！）

## 原型的评估

原型评估与易用性测试相关。



## 原型风险

**原型发布的压力**

最大的风险在于项目干系人看到一个可以运行的可抛弃原型后得出了一个产品几近完成的可怕结论。

所以一定要清楚原型正式的作用以及目的，特别是所有涉及到原型的人员。

让所有人明白一个原型是不会作为产品软件交付发布的。所以其中一个控制原型的方法就是使用纸上模型而不是电子模型。

**受细节所累**

另一个风险在于用户把注意力放在UI有关的外观和操作细节上。应该重点关注的是与概念相关的需求问题，尽可能将原型限定于显示画面，功能和导航选项。

**不现实的性能预期**

第三个风险在于用户根据原型的性能来判断最终产品的预期性能。不要在预期产品环境中对原型进行评估。原型设计和产品开发多方面上都有差异，不能相提并论。

**对原型投入过多**

最后，不要在原型工作上投入过多的精力，导致开发团队最后不得不没有时间而将原型作为产品或者匆忙开始开发。

原型能够测试假设，能够回答问题并且能够提炼需求就足够了！

## 原型成功的因素

