

《精益设计：设计团队如何改善用户体验（第二版）》

一.简介及基本原则

1.Lean UX：从未如此重要

- 设计总在不断演进
 - 敏捷软件开发、持续集成、持续部署
 - 能够迅速且持续地了解产品满足客户需求的程度
 - 提升客户对产品质量的期望及其对公司响应问题和反馈速度的预期
 - 成本不昂贵
 - Lean UX是产品设计和团队协作的演进
 - 提炼出设计工具中最精华的部分，并与敏捷软件开发及精益创业思维相结合
 - 更加接近客户的声音
 - Lean UX在于深度地跨职能协作
 - Lean UX改变了谈论设计的方式
 - 客观地谈论什么东西是有用的

2.基本原则

适用于设计流程、团队文化以及团队组织

- Lean UX建立的基础
 - 用户体验设计（识别系统用户的需求）
 - 敏捷开发方法
 - 个体和互动重于流程和工具
 - 可用的软件重于详尽的文档
 - 客户协作重于合同谈判
 - 应对变化重于遵循计划
 - 精益创业法
 - “开发——评估——认知”的反馈循环来最小化项目风险
 - 研发最小可行产品（MVP），迅速推向市场，及早开始认知的过程
- Lean UX的原则
 - 指导团队组织的原则
 - 跨职能团队
 - 小型、专注、聚合
 - 自给自足、自我驱动
 - 聚焦于问题的团队
 - 指导流程的原则
 - 从怀疑到确定
 - 要成果，而非产出
 - 消除浪费
 - 共识

- 不要明星、权威、独行侠
- 允许失败
- 指导文化的原则
 - 小批量，减轻风险
 - 持续探索
 - GOOB (getting out of the building) ——新的“用户中心”方法
 - 把工作具像化
 - 实干甚于分析
 - 从交付中走出来

二.流程

Lean UX的日常工作形式：整个团队协同合作，进行迭代式研发，大部分工作都同时进行，交接很少，需要交付的东西减到最低，关注可用的软件及市场的反馈。

1.成果驱动愿景

- 设想
 - 描述设想
 - 谁：所有成员
 - 准备工作：
 1. 现有产品使用情况的分析报告
 2. 阐明用户行为原因的易用性报告
 3. 过去解决问题的方法和经验教训
 4. 利益相关者的看法，即如果解决这个问题，会给公司带来什么样的效益
 5. 展示竞争者如何解决同样问题的竞争者分析报告
 - 问题陈述
 - 问题模版
 1. 产品或者系统的目标
 2. 利益相关者希望解决的问题（即某个部分未能达成目标）
 3. 对于没有指明具体解决方案的，提出明确的改善需求
 - 设想表格

业务设想	用户设想
1. 我认为我的客户有__的需要。	1. 我们的用户是谁?
2. 这些需要可以由__来解决。	2. 我们的产品如何融入他们的工作或者生活?
3. 我的首批客户是__。	3. 我们的产品用于解决什么问题?
4. 客户想从我的服务中获得的最大价值是__。	4. 用户会在什么时候用什么方式来使用我们的产品?
5. 客户还可以得到__等附加好处。	5. 哪些功能最重要?
6. 我将使用__作为主要客户获取渠道。	6. 我们的产品应该是什么样子，又该如何工作?
7. 我赚钱的方式是__。	
8. 我的主要竞争对手是__。	
9. 我们能打败他们的原因是__。	
10. 我们最大的产品风险是__。	
11. 我们解决这个问题的方法是__。	
12. 还有哪些假设，一旦不成立就会导致项目或产品失败? __。	

- 为设想排序：假设的风险越高，就越应该优先验证
- 假设
 - 对假设进行细分
 - 【某些人群/某个人物性格】来【这样做/开发某种功能/提供某种体验】可以让我们获得【某种成果】
 - 验证方式是【市场是否有某种反馈/某个量化指标/某个定性指标】
 - 挑选参照物
 - 完成假设陈述
- 成果
 - 清晰地记录取得的成果
- 人物性格
 - 人物原型
 - 人物性格格式
 - 画像和名字
 - 影响行为的基本信息
 - 痛点和需求
 - 潜在的解决方案
 - 人物性格制作
- 功能

- 功能是服务业务、客户和用户的
- 把上述要素串起来
 - 头脑风暴

2.协作式设计

- 密切的跨职能协作的好处
 - 为什么每个人都要参与到设计中来
 - + 为什么低保真的草稿可以促进合作
 - + 如何让整个团队达成共识
- 高效协作设计的方法
 - 设计工作室，即全团队一起参加的协作草绘练习
 - 步骤
 1. 明确问题和限制条件——15~45min
 2. 每个人独立思考解决方案（发散）——10min
 3. 解决方案展示以及评论——3min/人
 4. 迭代并改善（呈现）——5~10min
 5. 团队一起思考解决方案（收敛）——45min
 - 风格指南和模式库，也就是即时可用的产品UI库
 - 制作风格指南
 - 一蹴而就法
 - 水滴石穿法
 - 维护风格指南
 - 风格指南包括的内容
 - 所有屏幕上显示的东西，表单填空框、标签、下拉菜单、单选按钮、UI行为、Ajax、jQuery等
 - 所有的互动设计元素，外观、常用摆放位置、什么时候使用
 - 视觉设计元素，颜色组合、不同状态时的不同颜色、标示、页首、页尾、格子系统、字体的选择等
 - 文案的协作模式，语言风格、当用和禁用的词汇、语法的选择、当用和禁用的口语、按钮使用的语言（确定、OK、同意等）、导航语言（上一步、下一步、显示更多等）
 - 成果的风格指南的特点
 - 容易获取
 1. 容易找到
 2. 渠道通畅
 3. 可以搜索
 - 持续改善
 - 易于采用
 - 制作风格指南的步骤
 1. 制作目录
 2. 添加素材
 - 分布在不同地域的团队的合作方法

3.最小可行产品（MVP）及运用

- MVP

- 概念

- 一个能验证所有假设的最简产品

- MVP的专注点

- 增进认识：探索市场需求
 - 价值传递：制作简易版的产品或功能，尽早为客户创造价值

- 创建MVP

- 针对增进认识

- 简明扼要：提炼想法，留下核心价值主张
 - 排序要狠：抓住最有价值的点子
 - 保持敏捷：使用可快速修改和更新的方法来设计
 - 评估行为：直接观察和客户的行为而不是语言
 - 使用CTA：行为召唤——要求用户执行一个特定的动作来判断用户是否感兴趣

- 针对价值传递

- 能派上用场：把需要测试的功能和产品的其他部分联系起来，构建出更真实的使用场景
 - 和现有系统对接：在现有产品 workflows 中执行
 - 和谐一致：使用现有风格和品牌指南来设计新功能

- 原型

高效的MVP工具，让用户感受到接近最终产品的使用体验。必须是可点击或接触的。此外需要选择适合的工具来尽可能花最少时间实现它。

- 原型工具

- 低保真原型：纸质原型

- 优点：

- 1. 不到1小时就能做出来
 - 2. 组合和重组都很容易
 - 3. 成本低
 - 4. 材料在办公室就有
 - 5. 大部分人觉得好玩

- 缺点：

- 1. 快速地迭代和复制原型费时费力
 - 2. 模拟的感觉非常假
 - 3. 能得到的反馈仅限于高层级的信息结构或工作流程

- 低保真原型：可点击的线框图

- 优点：

- 1. 让人对工作流程有更直观的感受
 - 2. 发现用户在执行主要任务时的重大障碍
 - 3. 验证核心元素的可见度
 - 4. 可利用现有元素快速创建出可点击的原型

- 缺点：

- 1. 使用者心里明白是半成品

- 2. 人们会格外注意标签和文字
- 制作工具
 - 1. Balsamiq
 - 2. Visio
 - 3. OmniGraffle
 - 4. PowerPoint
 - 5. Fluid Designer/Pop Prototype on Paper
- 中高保真度的原型
 - 优点：
 - 1. 可生成高品质的真实原型
 - 2. 可测试品牌元素和视觉设计效果
 - 3. 可测试工作流程和界面互动元素
 - 缺点：
 - 1. 和原生原型相比互动性仍有欠缺
 - 2. 通常无法用真实数据来进行互动，所以可模拟的互动有限
 - 3. 耗时且维护麻烦
 - 制作工具：
 - 1. Axure RP
 - 2. Adobe Fireworks
- 代码型原型
 - 优点：
 - 1. 可以在生产环境中重用代码
 - 2. 最真实的模拟效果
 - 3. 可以从现有代码中生成
 - 缺点：
 - 1. 团队可能会纠结于原型的细节
 - 2. 需要花费很多时间才能做出想要的体验
 - 3. 在发布之前会有把代码写完美的冲动
 - 4. 升级和迭代需要花费很多时间
- 原型应该专注于核心工作流程
 - 非原型类MVP
 - Email
 - Google Adwords
 - 着陆页面
 - 无用按钮

4.反馈和研究

- 持续和协作式研究
 - 协作式探索：整个团队一起通过市场来验证某个想法
 - 实地协作式探索
 - 大家一起评审问题、设想、假设以及MVP，并一起决定需要了解什么
 - 大家一起决定找谁来谈才能了解到我们需要了解的东西

- 做一个访谈指南，作为访谈的方法指导
 - 把团队分成访谈结对小组，每个小组尽量包含不同的角色
 - 每个访谈结对小组手上应该有一份MVP
 - 每个小组分别出击，和客户或用户会谈
 - 一个成员做访谈，另一个做记录
 - 先问问题、聊天，并观察对方的反应
 - 在访谈的后半段再展示MVP，并让客户自己适用一下
 - 收集客户的反馈
 - 访谈完成之后，成员交换角色，让做记录的人来问后续的一些问题
 - 访谈结束时，请客户推荐其他可能提供有用反馈的人
- 持续探索
 - “3-12-1”方法：3个用户、中午12点、每周1次
- 案例研究
 - 团队一起分析研究成果
 - 迷惑、矛盾和马马虎虎之处
 - 寻找规律
 - 把异常数据放好
 - 通过其他信息来源进行验证
 - 时间会让规律浮现
 - 所有东西都需要测试
 - 草图
 - 静态线框图
 - 高保真视觉原型
 - 互动是原型
 - 代码原型
- 持续协作式探索的考察技巧
 - 客户服务
 - 请教客户对于目前正在研究的那部分产品有什么意见
 - 每月和客户碰一次面，了解目前的形势
 - 了解客户会如何解决目前遇到的问题。让客户参与到设计流程和评审中国呢来
 - 在客户询问相关问题是提出假设的解决方案
 - 内置反馈问卷
 - 问卷形式
 - 邮件表单
 - 客户支持论坛
 - 第三方社区网站
 - 研究反馈结果
 - 搜索记录
 - 网站使用情况
 - A/B测试

三.在组织中使用Lean UX

1.把Lean UX与敏捷方法相结合

- 术语
 - Scrum
 - 一种敏捷方法论，推崇现实迭代，自行组织，具有高度团队责任感
 - 用户故事
 - 作为一个用户，我希望能做到某种事情，这样我能获得某种好处
 - 任务储备
 - 一系列已经按照优先级排序的用户故事
 - Sprint
 - 一个Sprint就是一个迭代
 - 站会
 - 每天都会进行的短会，说明当天做的事情
 - 回顾会议
 - 迭代安排会议

- 超越交错式Sprint

设计比开发提前一个sprint，所有开发任务都在“设计sprint”中进行设计和验证，然后任务分配给开发团队，由他们在“开发sprint”中进行开发。

使用这种方法可以让设计得到更多验证，让团队学会了如何把迭代做到更小，如何把大的目标拆分成连续的小任务。

同时，这种方法会让整个团队容易分裂成两方，无法享受到跨职能协作团队的好处，造成不必要的浪费。

- 配合Scrum的节奏来实施Lean UX
 - 主题：几个sprint组成一个主题
 - 构思启动短会
 - 迭代计划会议
 - 用户验证计划
- 全体参与
- 积极交流

2.组织层面的转变

- 成果式团队
 - 专注于想要取得的成果
- 分工
 - 能力大于职位
- 用户体验设计师的新技能
 - 设计师必须开放设计流程
 - 设计师必须具备领导能力
- 跨职能团队
 - 平等互动

- 小团队
- 工作空间
 - 每个团队成员都能看到其他人
- 拒绝英雄主义
- 速度第一，美化第二
- 推崇解决问题的思维
- UX孽债
 - 持续改善用户体验
- 以交付为目的的设计公司
 - 加强客户和设计公司之间的协作
 - 把重心放在成果而不是交付上
- 第三方软件开发团队一起工作
- 文档标准
- 考虑现实环境
- 自下而上、由内向外的管理模式

3.研究案例

- 监管和金融服务：PayPal
 - 优化结算流程
 - 团队
 - 启动项目并克服困难
- 从线上到线下：CarMax
 - 寻求成果
 - Lean UX + 客户体验 + 服务设计
 - 人物原型
 - 检验假设
 - 下一次迭代
 - 检验其他假设
 - 整合实体店销售人员
 - 团队的日常节奏
- 为客户设置预期：ustwo
 - 服务定义工作室
 - 工作室之后MVP及协作
- 转变营销方式：数字广告公司
 - 结盟、协调、灵活
 - 与第三方开发人员合作

四.总结

Lean UX有三层含义：1.是设计师和产品团队工作流程的变革；2.是让我们以一种谦卑的方式对待工作的文化变革；3.是一种对软件设计及开发团队进行组织和管理的方式

在我看来：不仅仅是软件开发，其他行业也充满了Lean UX的身影，大家实行的步骤、方式也不尽相同。因此不同领域、不同规模的公司/团队需要探索一套适合自身并可行的方案。往往并不是所有方法都必要（可行），根据指导思想针对性删减流程，寻求利益平衡才是需要做的。

此外，我不再讨厌一直开会了。当然，是那种头脑风暴的讨论会。