



**本科毕业设计**

外文参考文献译文及原文

**学 院 环境科学与工程学院**

**专 业 环境科学**

**年级班别 2015 级（1）班**

**学 号 3115006378**

**学生姓名 黄健楸**

**指导教师 王孝武**

**2019 年 5 月**

目录

[4 移动设计流程 1](#_Toc9954436)

[4.1 观察现实中的人与手机的交互 1](#_Toc9954437)

[4.2 你的原型方法必须能适应多样的物理形态因子 1](#_Toc9954438)

[4.3 用户测试必须允许人们对自然范围内的动作、声音和多点触碰的探索 2](#_Toc9954439)

[4.4 接触界面应体现简洁和精致 2](#_Toc9954440)

[4.5 强制愉悦 3](#_Toc9954441)

[4.6 讲一个完整的故事——跨平台体验的设计 3](#_Toc9954442)

[4.7 移动设计案例学习 4](#_Toc9954443)

[4.7.1 第一步：范围、概念和计划 4](#_Toc9954444)

[4.7.2 第二步：设计研讨会 6](#_Toc9954445)

[4.7.3 第三步：便利贴上的示意图和RITE 7](#_Toc9954446)

[4.7.4 第四步：视觉设计 12](#_Toc9954447)

[4 Mobile Design Process 13](#_Toc9954448)

[4.1 Observe Human-Mobile Interaction in the Real World 13](#_Toc9954449)

[4.2 Your Prototyping Methods Must Allow for Variety in Form Factors 14](#_Toc9954450)

[4.3 Your User Testing Must Allow People to Explore the Natural Range of Motion, Voice, and Multitouch 14](#_Toc9954451)

[4.4 Touch Interfaces Embody Simplicity and Sophistication 15](#_Toc9954452)

[4.5 Delight Is Mandatory 16](#_Toc9954453)

[4.6 Tell a Complete Story—Design for Cross-Channel Experiences 16](#_Toc9954454)

[4.7 Mobile Design Case Study 17](#_Toc9954455)

[4.7.1 Step 1: Scope, Concept, and Planning 17](#_Toc9954456)

[4.7.2 Step 2: Design Workshop 20](#_Toc9954457)

[4.7.3 Step 3: Wireframe and RITE Study with Sticky Notes 22](#_Toc9954458)

# 移动设计流程

是什么组成高效的移动设计流程呢？在本章的末尾，我描述了一个端到端移动设计过程案例学习，它展示了贯穿本书的"便签“移动设计方法论。但在你跳到这个小节之前，我想先谈论移动时代设计的一些挑战，以及将传统的以用户为中心的技术移植到的移动端的一些方式，从而让他们保持高效而重要。

* 1. 观察现实中的人与手机的交互

在过去，在软件设计的时候，情景（Context）是要考虑的因素之一，但它常常沦为其他分析因素的替补。为什么呢？因为在移动设备到来（arrival）之前，情景通常就是一台用户坐在电脑前，除非你在设计一台电脑控制的咖啡机（这笑话可真冷）。因此，情景，即“在与你的软件交互时”，你的用户基本就坐在电脑屏幕前，手上有着键盘和鼠标。

而现在，在移动设计的时候，情景为王。理想情况下，你和你的团队应该观察现实中的情景，获取第一手资料。因为现在再无可能仅靠想象与建模就能弄清交互是怎样进行的（一个人坐在电脑前，抓着鼠标……），因为人的行为以及她和移动设备的交互高度依赖于情景。就连设备方向、手抓方式（hand grip）等基本设计参数都随着场景在动态变化——用户站在繁忙的街角，看着地图时；和配偶坐在沙发上，分享孩子的照片时；一只手正忙着跟老板通话，另一只手准备停车时；或在城市客车上阅读时……为了真正理解发生的事情，你和你的团队必须亲临现场，在这些交互发生时，获取它们的第一手资料。还有，在现场问简单的问题已经不能获取精确的数据。为了做出可靠的设计决策，理想情况下，观察行为时，你应该使用你应用程序的真实原型作为来引发这些行为反应。

## 你的原型方法必须能适应多样的物理形态因子

Mac和PC操作系统之间旷日持久的斗争主导着技术领域，接着是浏览器间的。站在用户体验设计的角度上说，PC是跟Mac不同的东西；但站在用户的角度来讲，这两者也许没那么不同——它们都是有鼠标、键盘和大显示屏的电脑。再加上大多数软件是为浏览器搭建的，所以用户体验基本与设备无关。例如，雅虎和脸书，各自在Vistra和Mac OS X自带的浏览器上看起来差不多。

而移动触摸屏的时代相反，因为它催生了各种各样的平台以及设备的外形。小屏手机、大屏手机和小、中、大屏平板，如今都货源充足、可供出售。而因为人体工学、外形因素以及一般模式的使用（比如大屏平板的joint ownership）。这些区别已经在第三章“安卓片段”讨论过了，并会贯穿全书。

而手机和平板不是你只需要单单考虑的平台——滑雪护目镜、冰箱、汽车等都是可预见的能安装Android系统的平台。为了迎合这些设备的需求，需要大量更改用户界面。这意味着旧的线性模型不再反映(reflects) the rich and variable reality on the ground。为了理解这些物体带来的设计约束，你要不断调整你的设计去接近设备的物理形态和瞬态元素如动画和过渡。

## 用户测试必须允许人们对自然范围内的动作、声音和多点触碰的探索

当谈到移动设计和移动测试，要忘掉你所知道的与电脑交互的知识。只用鼠标键盘与电脑交互的单一模式不适用于移动设备。移动时代的大部分内容是运用人体自然运动的优势：刮屏幕代表更深入，晃动手机代表说不，以及将手机贴着耳朵打电话。从语音识别数字助手，到使用身体摇摆的计步器，再到GPS 车载传感器来决定日常运动的速度和量，如今的移动设备正运用前所未有的动作、声音和多点触碰手势，从用户那获得越来越复杂的输入。为了设计有效的界面，你的原型和用户体验测试技术需要考虑这些与设备交互的全新模式。

## 接触界面应体现简洁和精致

在大屏幕上运行的电脑软件可以有考虑不周的广告模块，还能成功地保留用户。只因这里的用户不得不坐在椅子上用电脑，使得他们能高度容忍复杂性、能相对集中注意力到软件本身。

移动时代就是关于移动性的。那意味着你的用户的注意力比以往任何时候都分散，甚至比五年前任何人想象地都多。也就意味着界面必须简洁。 这不是说“简单”——就像爱德华·塔夫特(EdwardTufte)所说，简洁和头脑简单是有巨大区别的——正相反，软件需要负担从前可以让用户承担的复杂性。

不要误会：人们确实想在手机和平板上做更多的事情。你只是不能像之前一样提供那么复杂的东西。所以在接触界面，你有一个独特的用户界面，不能复杂却要非常精致。这意味着，设备的触摸界面在很多方面是比台式电脑的web界面要容易做出原型的——特别是涉及纸张原型等低保真度方法时——只要调试人员努力探索界面不太明显的方面。

## 强制愉悦

愉悦、娱乐和游戏存在于旧的PC与Mac系统中。然而，大部分“娱乐”只归于特定的活动，比如电脑游戏。新的移动平台是在游戏中成长起来的——它的血液流淌着游戏，它的DNA与游戏同源。因此，不论任务多无聊而琐碎，设计者都要确保软件让人愉悦，虽然“愉悦”可能只是意味着这个软件能帮助用户尽快完成任务。

进一步游戏化是新平台的自然结果，正如约翰·费拉拉（John Ferrara）在他的书《好玩的设计》(Rosenfeld Media, 2012)中所说，游戏必须是一种令人愉快的独立活动，而不是被附加到其他日常事项上。这意味着最好的移动体验必须更像游戏。例如，屏幕的小尺寸不可避免地使得有趣的元素(如过渡)在体验中扮演重要角色。相比之下，旧的浏览器模型只有很少的过渡。这意味着，当你在为应用设计原型时，你必须留出时间去探索过渡、愉悦和游戏这些元素。

## 讲一个完整的故事——跨平台体验的设计

使用PC或Mac几乎总是可行的。大多数人会用“在线”或“电脑”时间来完成数字任务，除了几百个超级极客，他们把电脑带到洗手间，拒绝洗澡，从而尽可能呆在线上。而更多“正常”的人总是随身携带着他们的移动设备。越来越多的人带着数码设备睡觉、吃饭，甚至上厕所(令人震惊!)。由于不可思议的机载传感器阵列(麦克风、GPS、光传感器、相机、近场通信[NFC]、触摸、运动，等等)，移动体验在离线(也称为“现实世界”)和虚拟世界之间创造了一种前所未有的连接。这就好像我们获得了一个新的器官，它让我们连接到看不见的Facebook的数字世界，扫描二维码和NFC芯片，并在我们需要信息的时候访问相互关联的数字信息，比如地图和评论。这种新的“移动器官”一直伴随着我们，以一种完全不同的方式来方便、快速地访问和处理信息。

如今，可以肯定的是，“移动器官”将被用来增加每一个传统的线下体验，比如参观游乐园、购物旅行，甚至是在树林里远足。作为一名设计师，您需要密切关注用于移动设备与其他平台交互的空间。例如，某人可能在移动设备上启动任务，却在桌面和社交网络上执行任务，然后在实体店完成任务。这些看起来是“在一旁”完成的快速任务，或者是在等待其他事件发生时临时完成的任务，可能只是您的主要移动用例。

现在，您已经了解所面临的一些挑战，下一节包括一个移动设计案例研究，它将帮助您理解如何将这些信息组合到一个适用于移动的以用户为中心的设计(UCD)过程中。

## 移动设计案例学习

我使用了一个轻量级的敏捷移动设计流程来帮助实现一个根本性的创新：一个真正的移动“60秒清单”流程，用于ThirstyPocket iPhone App，它将很快进入Android市场。本项目是一个关于如何将UCD应用于移动设计的说明性案例研究。本案例研究仅为了说明前面几节中讨论的一些概念，例如轻量级原型。你可能要根据具体情况来调整设计方法和流程。关键是保持灵活性，同时保持以客户为中心。

### 第一步：范围、概念和计划

在提出任何设计解决方案前，先开一个启动会议，来了解谁、在哪里、如何以及有多少问题——也就是情景、角色、愿景和预算。根据项目的不同，这可能简单地能用一句话来写。至关重要的是，任何问题都要提出，直到整个团队都同意这四点，这样才能通过调研得到答案。

#### 上下文：产品将在何处使用？

如上所述，情景是创造优秀移动设计的关键。你必须知道你的潜在客户在哪里，他们使用什么设备，他们在与你的应用程序交互时还在做什么，以及在这过程中他们的情绪状态和所担忧的事情。所有这些因素都将有助于推动设计，并最终决定你对最终设计方法的选择，以及您的产品应该具有的特性和系统行为的列表。在计划和范围确定阶段，您需要写下团队当前的想法，以作为客户研究的起点。对于ThirstyPocket APP，情景是“城市地区的车库销售”。

#### 角色：谁是目标用户？

可能你一开始就了解目标用户，也可能要从内部讨论或实地研究中推演出。不管你是如何得出这个洞见的，甚至可以不管正确与否，对此达成一致就好。如果团队中对目标用户特征有不同意见，那就把它记录下来——需要在有不同意见的地方多加研究。尽管推荐，但你不一定要花费大量时间为您的移动项目开发复杂、详细的角色。有时候，一句 “人物素描”就足以让我们投入到测试中去。就像ThirstyPocket APP只需一句“年轻大学生，没有很多钱或时间”。角色最重要的作用是凝聚团队和体会目标客户所面临的困难和挑战。如果你对没有很多信息的人物素描感到奇怪，请记住，有虚构的人物角色总比没有好。至少您将记录下您的团队的假设，这样您就可以快速发现这些假设是否正确，并在进行实地调研时根据需要进行更新。

#### 实地调查和情境访谈

在情景和角色素描完成之后，开始测试！为了ThirstyPocket，我们走访了大量的车库销售，并采访了符合销售目标的人，以便更好地了解现有销售系统和工作流程的挑战，以及人们对这个过程的不满。调研时，一定要和其他团队成员一起实地研究；调研结束后，要立即讨论调查结果并用头脑风暴产生想法。调研会议结束后，喝杯咖啡、吃顿午餐或晚餐，会有特别好的效果。不需要优秀的文档—使用草稿纸画图和与整个团队共享的想法图，通常是产生优秀想法、改进和产品愿景的最佳方法。不要忘记测试您的假设，并在必要时纠正角色素描和对情景的理解。

#### 愿景：用户会怎样使用产品？

你会怎样想象用户会怎样使用产品？会是长期交互，还是信息快餐？用户多久交互一次？是什么触发了这种交互？当客户与您的应用程序交互时，服务窗口是什么？交互是否为多点触碰，并且客户必须在稍后的某个时间点返回？使用该应用程序需要准备工作或培训吗？在前端，顾客之间会发生什么？在后端，软件或服务端会发生了什么？最终，作为设计过程的一部分，你要理解并打磨完整愿景。对于ThirstyPocket App，我们想出了一个很酷的口号：“列表只需60秒。”理想情况下，你对产品将如何使用的愿景应该源于对目标用户在目标情景中的观察。从这一点开始，很容易跳到下一节“第二步，设计研讨会”。

#### 预算：你计划在设计和开发上花多少时间和金钱？

用户体验设计只是产品开发的一个小部分。花时间了解设计如何与你的总体开发计划相适应，并按你的时间安排工作，以充分利用团队的技术能力。一个典型的设计过程需要三到六个月。对于ThirstyPocket应用程序，时间预算设定为六个月。

在建立了上下文、角色、愿景和预算之后，就可以进入下一节，设计研讨会了。

### 第二步：设计研讨会

在研讨会开始时，在您提出任何设计解决方案之前，您必须首先集中精力确定四个基本信息：角色、情景、场景和愿景。通过开发用例场景和增强的愿景声明，你将重点放在推动团队间共同的理解，这将帮助您填补在第一步中开发的框架中任何缺失的部分。在ThirstyPocket App中，用例场景和愿景更新如下：

* 场景：在大约五英里范围内卖一辆车;在音乐会的最后一分钟通过短信/电话进行交流来售票。
* 增强愿景：本地的、社交的、电子商务。亲自查看产品并使用现金支付，就像淘旧物一样。没有航运。只用一张照片。销售过程自然、简便、简单，且无需注册/登录。

在确定了四个要点之后，且常在这个过程中，整个团队已经产生了一些想法。

在头脑中处理多个方向的设计过程，并将严格的规则注入到头脑风暴的过程，会是一个好主意。与其对特定的方法感到兴奋，不如快速地记录下来，然后把它完全放在一边，提出以下问题：

* “有没有其他的方法来设计这个？”
* “如果你从X而不是Y开始呢？”
* “你能让它更像X吗？”
* “你能使新的工作流程适应现有的客户行为吗？”
* “你能让这个工作流程更像一个游戏吗？”

故事板是一种优秀的技术，可以记录各种设计思想在情景中以及随着时间的推移而发生的情况。关键是将实际的界面设计部分降到最低，而是描述应用程序将如何在移动情景中使用。对于ThirstyPocket，图 4.1中所示的方法非常适合这两种场景，并且实现了更大的产品愿景。

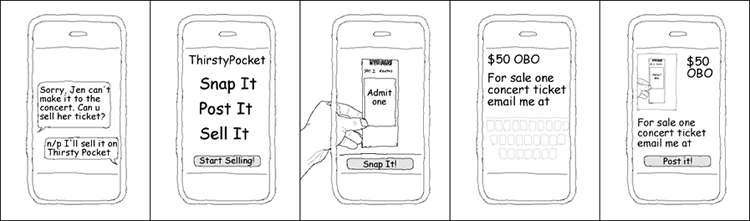


图 4.1这是ThirstyPocket应用程序销售过程的故事板

例如，故事板记录了以下场景：一个名叫Gene的年轻人和一群朋友坐在一辆车里去参加一场很酷的音乐会。突然他收到一条短信——他的朋友珍来不了！吉恩能卖她的票吗？“没问题，”吉恩回答说，“我可以在ThirstyPocket上卖。”这个开放的故事板很重要，因为它显示了交互的情景。它应该传达一些关于场景的东西，以及在场景发生的地方建立一种强烈的感觉。

接着，Gene启动了ThickPocket app(“Snap It，Post It，Sell It！”)，这个动作打开了内置的相机。然后吉恩轻点“Snap It！”按钮拍下他想卖的票的照片。吉恩填写了一个简要描述并点击“Post It!”按钮在某预览界面上。

这就是那个简单、快速、专注于当地的销售流程，且没有大量的界面细节。

在研讨会期间，您将专注于速度，快速探索白板上或带有小型矩形便笺的各种设计场景故事板。你必须先取得相互理解和强力愿景，而不是详尽的文档。对故事板技术的完整描述超出了本书的范围。查阅Making Comics (Harper, 2006) ，一本Scott McCloud所著的绝妙书籍，并访问这本书的配套站点http://AndroidDesignBook.com，了解更多移动故事板和用例。

在研讨会期间，鼓励团队中的每一个人参与其中并画画。高质量的产品故事板是不必要的。如果有一个布满潦草字迹的白板，就足以让整个团队“获得”特定的移动场景，也就完全达到了想要的效果。在完成了关键用例场景的故事版后，就到了“第三步：便利贴上的示意图和RITE”的时候了。

### 第三步：便利贴上的示意图和RITE

正如本章前面所讨论的，由于移动设备不寻常的设计限制，通常使用的UCD过程，即创建计算机生成的示意图，然后构建高保真度的原型，并不总是适用于移动设备设计。

与其花费大量的时间和精力创建高保真的示意图图，不如建立一个廉价而高效的快速迭代测试和评估(Rapid Iterative Testing and Evaluation，简称RITE)研究作为设计过程的核心。在设计过程中尽早进行礼仪研究，你就能在比你想象的更短的时间内创造出更令人愉快、更有用、更成功的移动产品。

在快速迭代研究中(如果你愿意，你可以称之为“快速迭代测试”，不过我更喜欢用“研究”这个词来强调设计的变化)，我通常推荐选用9到12名参与者，分3到4轮进行(每轮3名参与者)。快速迭代研究的关键部分是允许在两轮测试之间有时间更新原型，以修复前一轮测试中发现的问题。RITE研究基本上是一系列的设计/测试组合，根据客户、工程师和管理层的反馈，根据需要快速更改原型。

多年来，RITE研究一直是UCD工具箱的一部分。为了让RITE能够很好地作为移动设计流程的核心工作，我提出了一个简单的修改，即用便利贴制作原型。

便利贴原型提供了许多优势。首先，一大叠便利贴(我更喜欢3×5英寸大小)的尺寸与典型的手机相似。这意味着不需要创建任何外部案例、盒子或任何类似于移动设备的东西;这叠便利贴是一个完整的解决方案。

便利贴原型价格低廉，易于创建，而且相当健壮。它们可以从任何高度掉落，而不会解体，甚至不会分裂成单独的页面。在街上或在咖啡店，将一叠便利贴交给一个完全陌生的人，问他一些关于你的应用程序的问题，不会有什么问题 (大多数人不会有相同的轻松感受将他们宝贵的新型手机交给一个完全陌生的人)。如果参与者不小心把“手机”掉在地上了，它不会损坏，如果他们不小心把“手机”拿走了，你只会损失大约一美元!

如图 4.2所示，一叠便利贴与移动设备的形状因素非常相似，因此这个简单而精致的原型使你能够测试自然的人体工程学、多点触控和加速度计运动——这是传统示意图无法完成的。

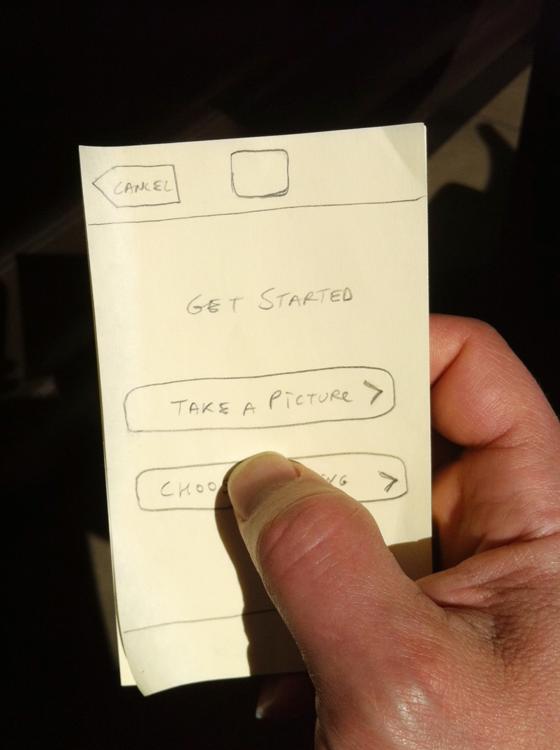


图 4.2使用一组便利贴来模拟手机是一种有效的轻量级原型技术

修复便签本原型的问题是很容易的：如果你发现了设计上的问题，你可以当场修复，用橡皮擦和铅笔，或者用一个新的便签本，用一个精细的永久性记号笔重新绘制。同样，如果您想测试一个可选流程，您可以在几分钟内绘制一个新的屏幕设计，并在将其提交给下一个评估人员时，几乎立即将其性能与现有的想法进行比较。这种易用性有助于您的设计快速发展，特别是如果您的整个核心团队都参加了这些测试会议。

图 4.3演示了如何用便利贴方法为ThirstyPocket测试的 “在60秒内销售您的产品”原始流程的示意图。

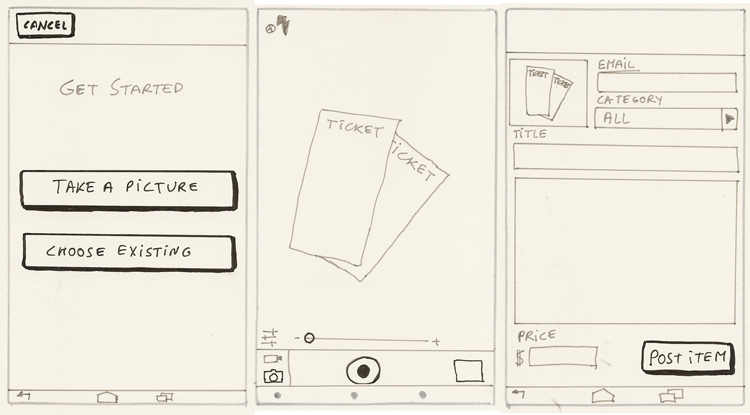


图 4.3用便利贴创建的用于早期测试的移动设计原型示意图

因为我个人在使用我的便签纸原型方法的过程中获得了很好的效果，并且我坚信这将帮助你达到设计目标，所以我为本书中的大多数设计模式创建了3×5英寸的便签纸示意图。当你探索这种原型技术时，你需要记住以下几点：

* 当使用便利贴模拟移动设备，不需要为便利贴加一个额外的框来表示这是屏幕。为了节省时间，也为了让绘图更容易理解，直接假设便利贴就是手机屏幕。对于安卓手机，您应该在适当的地方添加设备的硬件按钮(返回键、Home键等)。
* 为了保证易读性，这本书中的所有图纸都是黑白的，使用带有档案墨水的Pigma Micron画的。在真实原型中，我使用一支简单的2号自动铅笔，以便我可以立即擦除或更新“屏幕”上的任意元素。你可以用你喜欢的钢笔或铅笔，以及颜色。
* 我在画线框的时候会充分利用尺子。这是因为画一条直线可能会有点困难，特别是在这方面快速变化的时候。不管你是否决定使用尺子，最好在整个原型中保持一致，以免分散参与者的注意力。有各种各样的模板和其他绘图辅助工具提供帮助，但我就喜欢用透明的小三角尺。记住：关键不是最好的图纸，而是测试想法最快而又最高效的方式。最简单的绘图过程是最好的。用那些你用得最上手的和能最有效地表达设计的。
* 粘注原型可以很容易分支。这是通过只给评估者只有第一个“屏幕”卡在顶部的粘笔记包。一旦评估员点击一个按钮或在屏幕上执行一些功能，我可以选择适当的下一个屏幕，以显示出包的替代品，我在我手中，从评估员的看法。这样，测试可以非常现实：如果下一个评估者使用不同的控件，他会收到不同的屏幕，允许测试分支和绕过的工作流、回溯和其他现实生活行为。原型的这种能力产生丰富和健壮的行为数据。
* 一些实践之后，便利贴原型也可以用于测试过场。如果过场对你的交互很重要，试着用“slipping in”代替“让下一页从底部出现。”你觉得怎么样?”如果参与者的回答得并不不明确，比如“很好”，就问是否更喜欢一个不同的过场，或者问其对这一过场动画理解。你也可以在单独的便签上画出个别的过渡状态（见“Storyboarding iPad Transitions”，Boxes and Arrows, January 5, 2011，http://www.designcaffeine.com/articles/storyboarding-ipad-mobile-transitions/）。
* 通过另一张较小的便利贴，很容易就能模拟键盘，可以贴到参与者拿着的便利贴“手机”的顶部。这样，在测试期间的任何时候，任何屏幕都可以动态地转换到“键盘输入”状态。这种技术为原型提供了额外的灵活性，同时消除了重复绘制键盘的繁琐工作。
* 记得重温你的故事板。示意图的工作流应该遵循产品愿景。例如，比较图 4.1与图 4.3就能看出，示意图是原始愿景通过添加细节和界面元素自然而然地拓展出来的。例如，在我开始测试后不久，就发现人们有时想先拍一堆照片，然后再卖东西。因此，我添加了另一个界面，给客户一个选择：拍照或选择现有（图片）（见图 4.3）。这种改变很常见。这就是你要先在纸上测试想法的原因!如果在RITE研究中领会的新洞见之后，界面变化足够大，你也必须更新愿景故事板中对应的地方。而在我们的例子里，新增的页面没有改变基本的故事板场景——它只是增加了它的保真度——所以没有必要改变原来的愿景故事板（见图 4.1）。

便利贴原型能够快速且廉价地探索多种设计方法，同时不用精心制作的相机和其他小配件。便利贴原型也使你和你的整个团队能够脱离办公室的限制，大胆地去移动交互实际发生的地方：咖啡店、繁忙的街角、出租车和地铁。在你试图设计的交互实际发生的地方，把你的便利贴原型放在潜在用户的手中，有着巨大的价值。

同时，参与者可以轻松地通过头脑风暴产生有价值的想法，因为原型没有标准答案。在用户会使用你作品的地方，和他们一起开设计会议。这些会议产生的宝贵见解，你可以直接纳入你的设计，这比你想象的要快得多。这就是为什么我在书中加入了几乎每一种设计模式的手绘便利贴示意图。希望这些将予你灵感，让你建立自己的贴纸原型，并直接到目标客户所在的情景测试。如果你需要帮助，请访问这本书的配套网站（http://AndroidDesignBook.com），在那里我使用便利贴原型、移动使用案例故事板和许多其他资源来主持RITE研究实践的视频，这些资源旨在帮助你最大限度地利用这项强大的技术。

### 第四步：视觉设计

RITE 研究后并不能直接交付，它只是设计过程中的一个关键步骤。思考您的设计过程的最佳方法是，原型和交付的状态应该反映产品完成时的总体状态。这就是为什么在这个过程的早期，通过使用一个轻量级的设计过程，专注于为用户设计大致流程和屏幕布局，你就能快速推进项目。在大致流程被制定和测试之后，是时候进入最后一步了：视觉设计。

传统的做法是聘请视觉设计师和内容经理包装产品，让产品更有吸引力，并完成项目最后的部分。虽然视觉设计不是本书的重点，但第一和第二章包括一些关于这个主题的基本要点。

值得注意的是，视觉设计可以增强和减损交互设计的意图。有时视觉设计会产生很大的影响。因此，尽可能多地测试应用程序以确保最终版本仍然符合原来的视觉故事板的简单和优雅，即便经过设计地演变。风格可以是建立（或破坏）情感联系的关键，所以它也值得测试。

要运行最终测试，只需将测试设备（要小心!）交给你感兴趣的人，也许是在咖啡店排队等候的时候，说：“让我知道你对这个应用程序的看法，我请你喝咖啡。”这次五到八人的最终测试应该不超过一个小时。这是一个很好的最终测试：理想情况下，整个流程所花的时间不应超过人排队买咖啡的时间，而界面应该是吸引人的且不言自明的，足以让人在没有咖啡因的状态下使用。

# Mobile Design Process

What makes an effective mobile design process? At the end of the chapter, I describe an end-to-end mobile design process case study that showcases the sticky-note mobile design methodology used throughout the book. However, before you jump to this section, I want to discuss the challenges of designing in the mobile age and some approaches to adopting the classic User-Centered Design (UCD) techniques to the new medium so they remain effective and relevant.

## Observe Human-Mobile Interaction in the Real World

In the past, context was one of the considerations when designing software, but it often took the back seat to other methods of analysis. Why is that? Because, ladies and gentlemen, before the arrival of mobile devices, the context was a computer that the customer would sit in front of, unless you were designing a computerized coffee maker (vile abomination!). Thus at the moment of interaction with your software, your customers were basically sitting down in chairs in front of computer screens with keyboards and mice.

In contrast, in mobile design, context is king—context that you and your team should ideally observe first hand, in a real world. It is no longer possible to reliably imagine and model how the interaction would proceed (a person sits in front of the computer, grabs the mouse, and so on) because the person’s behavior and her interaction with the device is highly dependent on context. Even fundamental design parameters like device orientation and hand grip change dramatically when the customer is standing on a busy street corner and looking at a map; sharing photos of the kids while sitting on the couch with a spouse; talking with one’s boss one-handed while trying to park the car; or reading while riding the city bus. To really understand what happens, you and your team have to get out there to observe these interactions first hand, as they happen. And while you are out there, simply asking questions is no longer enough to get accurate and precise data. To make solid design decisions, ideally you should observe behaviors using a realistic prototype of your app as a tool for eliciting these behavioral responses.

## Your Prototyping Methods Must Allow for Variety in Form Factors

For many years, the same old fight between Mac and PC has dominated the tech landscape, followed by the browser wars. From the standpoint of the User Experience (UX) design, the PC was somewhat different from the Mac, but from the standpoint of the customer, the two were perhaps not that different—both operating systems used a mouse, keyboard, and large screen. Also, because the majority of software being produced was built for the Internet browser, the experience was largely device-independent. Yahoo! and Facebook looked similar in Internet Explorer on Vista and in Safari on Mac OS X.

In contrast, the age of mobile touch computing yielded a tremendous variety of platforms and device form factors. Small phones, large phones, small tablets, mid-size tablets, and large tablets are widely available for sale today, and all demand different approaches to software design because of ergonomics, form factors, and general patterns of use (such as joint ownership of larger tablets). These differences are discussed in Chapter 3 “Android Fragmentation,” and are considered throughout this book.

However, phones and tablets are not the only platforms you need to worry about. Futuristic installations of Android OS in everything from ski goggles to refrigerators and cars are only a matter of time. All these installations require considerable changes to the user interface to fit the specific needs of the device. That means that the old model of wireframing no longer reflects the rich and variable reality on the ground. In order to understand design constraints for these objects, you need to modify your design approach to include the physical form factor of the device, as well as transient elements such as animations and transitions.

## Your User Testing Must Allow People to Explore the Natural Range of Motion, Voice, and Multitouch

When it comes to mobile design and testing, forget everything you know about interacting with a computer. The uniform mode of interacting with a computer via only mouse and keyboard does not apply to mobile devices. Much of what the mobile age is all about is taking advantage of the body’s natural motions: scratching to dig deeper, shaking to say no, and bringing the phone to your ear to speak. From voice recognition digital assistants to pedometers that use the body’s swaying motion along with GPS on-board sensors to determine the speed and quality of daily physical activity, today’s mobile devices are using an unprecedented array of motion, voice, and multitouch gestures to obtain increasingly complex inputs from customers. In order to design effective interfaces, your prototype and the customer experience testing techniques need to account for a full range of these new modes of interaction with the devices.

## Touch Interfaces Embody Simplicity and Sophistication

Browser and OS-based software that runs on large screens could afford to have ill-conceived advertising modules while still succeeding in converting customers because of the high tolerance of complexity, fairly large screens, and relatively high degree of focus that customers had on your software just by virtue of having to sit down at a chair to use the computer.

The mobile age is all about, well, mobility. That means your customers’ attention is even more scattered than ever before, more, even, than anyone thought possible just five years ago. This means that interfaces need to be simple. That’s not to say “simplistic”—as Edward Tufte famously said, there is an ocean of difference between simplicity and simple-mindedness. Instead, the software needs to take on the burden of complexity that was heretofore acceptable to pass on to the consumer.

Don’t misunderstand: People actually want to do more with smartphones and tablets than they could do with the web. You simply can’t afford the perception of complexity anymore. So in touch interfaces, you have a unique customer interface that is not complicated, but very sophisticated. This means that a device’s touch interfaces are, in many respects, easier to prototype than desktop web interfaces—especially when it comes to low-fidelity methods like paper prototyping—as long as the moderator makes an effort to probe the less tangible aspects of the interface.

## Delight Is Mandatory

Delight, fun, and games existed under the old PC and Mac system. However, the majority of “fun” was devoted to specific activities, such as computer games. The new mobile platform has grown up on games. The games are in its blood and DNA. Thus, no matter how boring and trivial the task, designers need to make sure that the software is as delightful as possible, even if “delight” simply means the software helped the customer complete the task as quickly as possible.

Increased gamification is a natural outcome of the new platform, and, as John Ferrara says in his book Playful Design (Rosenfeld Media, 2012), the experience of play must be delightful as a stand-alone activity, not feel tacked on as an afterthought onto some other agenda. This means that the best mobile experiences must behave and feel more like games. The small size of the screen makes it inevitable, for example, that fun elements (such as transitions) play a big role in the experience. In contrast, the old browser model had minimal transitions. This means that while you are prototyping the design of your app, you must allow time to explore the elements of transitions, delight, and gamification.

## Tell a Complete Story—Design for Cross-Channel Experiences

Using a PC or Mac was almost always work. With the exception of a few hundred super-geeks who took their computers to the restroom with them and refused to shower so that they could spend as little time as possible offline, most people would have dedicated “online” or “computer” time to accomplish digital tasks. In contrast, many “normal” people have their mobile devices with them at all times. More and more, there are increasing numbers of people that sleep, eat, and even go to the restroom (shocking!) with their digital devices. Because of the incredible array of on-board sensors (microphone, GPS, light sensor, camera, near field communications [NFC], touch, motion, and so on) the mobile experience creates a completely unprecedented connection between offline (also known as “real-world”) and digital realities. It’s as though we have acquired a new organ that enables us to connect to the unseen digital worlds of Facebook, read QR codes and NFC chips, and access interconnected digital information such as maps and reviews in the moment we need the information. This new “mobile organ,” which is always with us, is a completely different way to easily and quickly access and manipulate information.

Today, it is safe to assume that the “mobile organ” will be used to augment every traditionally offline experience, such as a visit to an amusement park, a shopping trip, and even a hike in the woods. As a designer, you need to pay close attention to the spaces between interactions, where a mobile device is used to interact with other channels. For example, someone might start a task on a mobile device, continue on the desktop and social networks, and finish it in the physical store. These quick tasks that appear to be done “on the side,” at the spur of the moment, or while waiting for some other event to occur might just be your primary mobile use cases.

Now that you know some of the challenges you are facing, the following section includes a mobile design case study that will help you understand how you can put all this information together into a User-Centered Design (UCD) process that works for mobile.

## Mobile Design Case Study

I used a lightweight Agile mobile design process to help deliver a radical innovation: an authentically mobile “60-second listing” flow for the ThirstyPocket iPhone app, which will also be shortly coming to the Android Marketplace. This project is an illustrative case study of how to apply UCD to mobile design. This case study is provided merely as an illustration of some of the concepts discussed in the preceding sections, such as lightweight prototyping. You may have to adjust your design approach and process based on your particular situation. The key is to stay flexible while remaining customer-focused.

### Step 1: Scope, Concept, and Planning

Before proposing any design solutions, start by having a kick-off meeting to understanding the who, where, how, and how much questions, otherwise known as context, persona, vision, and budget. Depending on the project, this could be as simple as writing these in a one-sentence format. It is critical that the entire team agrees to these four points, and any questions are called out, so they can be answered through research.

#### Context: Where Will the Product Be Used?

As noted above, context is the key to creating a great mobile design. You must know where your potential customers are, what equipment they use, what else they are doing at the time of interacting with your app, and what emotional state and concerns they have about the process. All of these factors will help drive the design and ultimately determine your choice of a final design approach and the list of features and system behaviors your product should have. During the planning and scoping phase, you need to write down the team’s current thinking to have a starting point for customer research. For the ThirstyPocket app, the context was “Garage sale in an urban area.”

#### Persona: Who Is the Target Customer?

You may start out knowing this, or the idea of a target customer may evolve from the internal discussion or field research. Regardless of how you arrive at the insight, it is essential to have an agreement of who the solution is supposed to target, even if your assumption is not correct. If the team disagrees on the target persona, go ahead and document this—disagreements point to areas where more research is needed. Although recommended, you don’t need to spend a lot of time developing sophisticated, detailed personas for your mobile project. Sometimes a one-sentence “persona sketch,” such as a simple “young college student, not a lot of money or time” that we agreed upon for ThirstyPocket, could be enough to jump into testing. The most important function of the persona is the sense of team cohesion and empathy toward the struggles and challenges faced by the target customer. If you are feeling strange about coming up with a persona sketch without a lot of information, keep in mind that having fictional personas are better than none. At least you will have documented your team’s assumptions, so you can quickly discover if they are not correct and update as necessary when you jump into field research.

#### Field Research and Contextual Interviews

After the context and persona sketch is complete, jump into testing! For ThirstyPocket, we visited plenty of garage sales and interviewed people who fit the target profile about selling their stuff to get a better understanding of the challenges of the existing selling systems and workflows and people’s frustrations with the process. As you do your research, be sure to go on field studies together with other team members and discuss findings and brainstorm ideas immediately after the research sessions. This is especially great to do over a coffee, lunch, or dinner following the research session. No great documentation is needed—simple paper sketches and idea drawings shared in context with the entire team are often the best ways to come up with great ideas, improvements, and product vision. Don’t forget to test your assumptions and correct the persona sketch and context understanding as necessary.

#### Vision: How Will the Product Be Used?

How do you envision the product being used? Is it a long engagement or a quick information snack on the run? How often does the customer engage? What triggers the engagement? What is the service window when the customer interacts with your app? Does the interaction span multiple touch points, and must customers come back at some later time? Does your app require preparation or training? What happens front stage, between customers? What happens back stage on the software or service side of the equation? Ultimately, your job is to understand and sharpen your complete vision as part of your design process. For the ThirstyPocket app, we came up with a cool slogan: “List your item in 60 seconds.” Your vision for how the product will be used should ideally be rooted in direct team observation of the target customers in target context. From this point, it is very easy to jump into the next section, “Step 2, Design Workshop.”

#### Budget: How Much Time and Money Do You Plan to Spend on Design and Development?

UX design is only a small piece of the product development puzzle. Take the time to understand how the design fits into your overall development plan and work within your timelines to make the most of your team’s technical capabilities. A typical design process takes anywhere from 3 to 6 months. For the ThirstyPocket app, the budget was set for 3 months.

After you’ve established your context, persona, vision, and budget, you can move on to the design workshop described in the next section.

### Step 2: Design Workshop

At the start of the workshop, and before you propose any design solutions, you must first focus on nailing down four essential pieces of information: personas, context, scenarios, and vision. You focus on driving the shared team-wide understanding and buy-in by developing use-case scenarios and a beefed-up vision statement, which help you fill in any missing pieces in the framework you developed in the first step. In the case of the ThirstyPocket app, use case scenarios and vision were updated as follows:

* Scenarios: Selling a car in the neighborhood, around the 5-mile range; selling tickets at the last minute while at a concert using texting/phone for communication.
* Enhanced Vision: Local, social, e-commerce. Seeing the product in person and paying in cash, like a garage sale. No shipping. Only a single picture. Natural, easy, and simple selling process, with no registration/login.

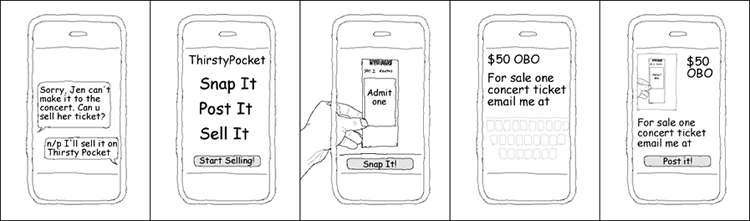
After you nail down the four essentials, and often during the process, you generate ideas by working collaboratively as a team.

It is a good idea to always approach the design process with multiple directions in mind, injecting a rigorous discipline into the brainstorming process. Rather than getting excited about the particular approach, document it quickly and then set it completely aside, asking the following questions:

* “Is there another way to design this?”
* “What if you start with X instead of Y?”
* “Can you make it more like X?”
* “Can you make the new workflow fit the existing customer behaviors?”
* “Can you make this workflow feel more like a game?”

Storyboarding is an excellent technique to document various design ideas in context and as they happen through time. The key is to keep the actual interface design portions to the minimum and instead describe how the app will be used in the mobile context. For ThirstyPocket, the approach shown in Figure 4-1 fit well with both scenarios, and it fulfilled the larger product vision.

Figure 4-1: This is the storyboard for the ThirstyPocket app selling process.



For example, the storyboard documents the following scenario: A young person named Gene is heading to a cool concert in a car with a bunch of friends. Suddenly he gets a text—his friend Jen can’t make it! Could Gene sell her ticket? “No problem,” replies Gene, “I can sell it on ThirstyPocket.” This opening storyboard is important because it shows the context of interaction. It should convey something about the scenario as well as setting up a strong sense of place where the scenario is happening.

Gene next fires up the ThirstyPocket app (“Snap it, Post it, Sell it!”), and proceeds to tap Start Selling! This action opens the built-in camera, and Gene taps the Snap It! button to take the picture of the ticket he is looking to sell. Gene fills out a quick description and taps the Post It! button on some kind of a preview screen. And there you have it—a simple, quick, locally focused selling flow without a great deal of interface details.

During the workshop you concentrate on speed, quickly exploring various design scenario storyboards on a whiteboard or with small rectangular sticky notes. You must strive first and foremost for mutual understanding and strong vision, not exhaustive documentation. A full description of storyboarding techniques is outside the scope of this book. Check out the incomparable Making Comics (Harper, 2006) by Scott McCloud, and visit the companion site for this book, http://AndroidDesignBook.com, for more examples of mobile storyboards and use cases.

During the workshop, encourage everyone on your team to draw and participate. High-value production storyboards are not necessary. If a whiteboard full of chicken scratches surrounding stick people is enough for the entire team to “get” the particular mobile scenario, that’s all you need. After you have the key use-case scenarios story-boarded, it’s time for “Step 3: Wireframe and RITE Study with Sticky Notes.”

### Step 3: Wireframe and RITE Study with Sticky Notes

As discussed earlier in the chapter, due to unusual design constraints of mobile, the commonly used UCD process of creating computer-generated wireframes and then building a high-fidelity prototype does not always work for mobile design.

Instead of spending a lot of time and effort creating high-fidelity wireframes, set up a cheap and efficient Rapid Iterative Testing and Evaluation (RITE) study as the core of your design process. Do RITE studies as early as possible in the design process and you will reap the benefits of creating more delightful, usable, and successful mobile products in less time than you ever thought possible.

The RITE study (you can call it “RITE test” if you like, though I prefer the term “study” to emphasize the design changes) I typically recommend is conducted using 9 to 12 participants in three to four rounds (3 participants per round). The critical component of a RITE study is to allow time between rounds for updating the prototype to fix the issues discovered during the previous round’s testing. Basically a RITE study is a series of design/test pairings where the prototype is rapidly changed as needed based on feedback from the customers, engineers, and management.

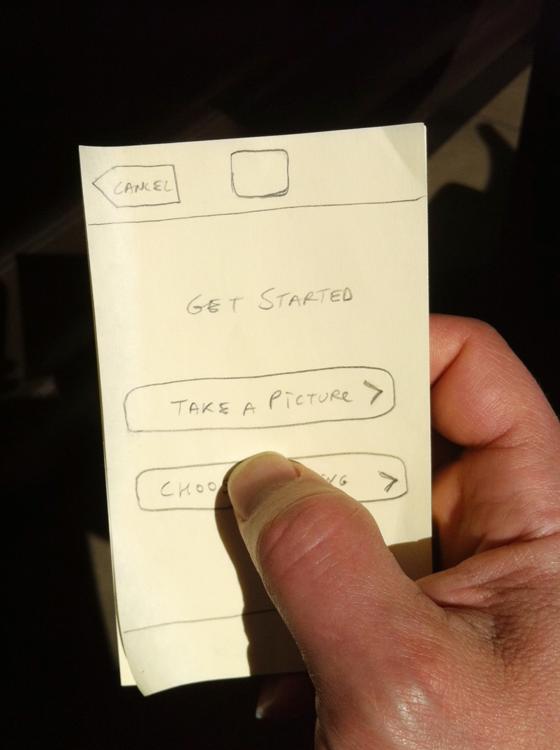
RITE studies have been a part of the UCD toolbox for many years. One simple modification I came up with to enable RITE to work well as a core of the mobile design process is to employ the prototype made from sticky notes.

Mobile sticky-notes prototypes offer many advantages. To begin with, a pack of large sticky notes (I prefer to use 3 × 5 inch size) has the dimensions that resemble those of a typical mobile phone. That means that there is no need to create any external cases, boxes, or anything else to resemble the mobile device; the pack of sticky notes is a complete solution.

Sticky-note prototypes are cheap, are easy to create, and they are fairly robust. They can be dropped from any height without disintegrating or even so much as falling apart into individual pages. You won’t have any issues handing a pack of post it notes to a complete stranger on the street or in the coffee shop and asking him some questions about your app (most people would not have the same light-hearted feelings about handing their precious late-model mobile phone to a complete stranger). If the participant happens to accidentally drop the sticky-note “phone,” it will not be damaged, and if they happen to run off with it, you will only be out about a dollar!

As shown in Figure 4-2, a pack of sticky notes closely resembles the form-factor of the mobile device, so this simple yet sophisticated prototype enables you to test the natural ergonomics, multitouch, and accelerometer motions—something traditional wireframes simply cannot accomplish.

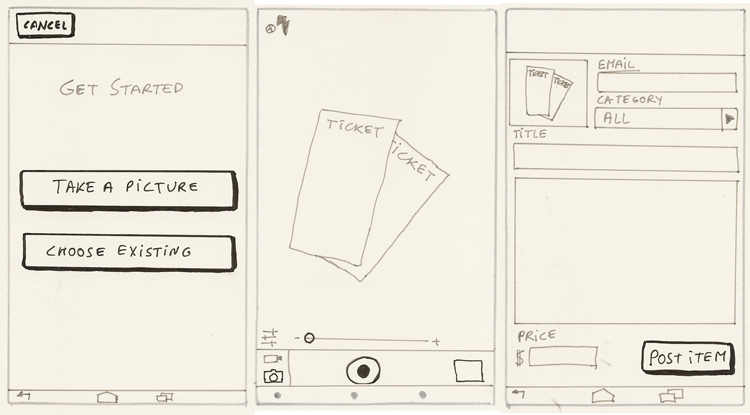
Figure 4-2: Using a pack of sticky notes to simulate a phone is an effective, lightweight prototyping technique.



Sticky-note prototypes are easy to fix: If you discover issues with the design, you can fix the interface right then and there, using an eraser and a pencil, or a new sticky note re-drawn with a fine-point permanent marker. Likewise, if you want to test an alternative flow, you can draw a new screen design in a few minutes and compare its performance with that of an existing idea almost immediately, when you hand it to the next evaluator. This ease of use helps your design evolve quickly, especially if your entire core team is present at these test sessions.

Figure 4-3 shows how I used the sticky-note method to wireframe the original “Sell your item in 60 seconds” flow I tested for ThirstyPocket.

Figure 4-3: This is the mobile design prototype for early testing that I created using sticky notes.



Because I have personally experienced excellent results using my sticky-note prototype approach, and I firmly believe that it will help you reach your design goals, I have created 3 × 5 inch sticky note wireframes for most of the design patterns in this book. As you explore this prototyping technique, here are a few things to keep in mind:

* When using the sticky-note pack to approximate the mobile device, you do not need to draw an additional box on top of the sticky note to represent the screen. To save time, as well as to make the drawing more understandable, go ahead and assume that the entire sticky-note surface is the screen of the mobile phone. For Android phones you should add the device’s hardware buttons (Return, Home, etc.) where appropriate.
* All drawings in this book are black and white, executed using Pigma Micron pens with Archival Ink. This is done for legibility. In real-life prototyping I use a simple mechanical #2 pencil so that I can immediately erase or update any elements on the “screen.” You can use any pen or pencil you like, black or color.
* I use a ruler liberally while drawing my wireframes. This is because drawing a straight line, especially while making rapid changes in the field, can be a bit of a challenge. Whether or not you decide to use a ruler, it’s best to be consistent in your technique throughout the prototype, so as not to distract your study participants. There are various templates and other drawing aids out there to help you. I am not a fan of anything other than a small transparent triangular ruler. Remember: The key is not the best drawings, but the fastest and most efficient way to test concepts. The easiest drawing process is the best. Use whatever works for you and conveys your designs most effectively.
* Sticky-note prototypes can be branched easily. This is done through handing the evaluator only the first “screen” stuck on top of the sticky-note pack. Once the evaluator taps a button or performs some function on that screen, I can select the appropriate next screen to show out of the pack of alternatives I hold in my hand, out of the view of the evaluator. This way the testing can be very realistic: If the next evaluator taps a different control, he receives a different screen, allowing for testing of branched and round-about workflows, backtracking and other real-life behaviors. This capability of the prototype yields rich and robust behavioral data.
* With a bit of practice, sticky-note prototypes can also be used to test transitions. If transitions are important to your interaction, try “slipping in” the next “screen” while saying something such as, “Let’s say the next page comes up from the bottom like so. How does that feel for you?” If the participant responds with something noncommittal such as, “Fine,” ask her if she would prefer a different transition instead, or if this movement has some sort of meaning to her. You might need to do a complex transition more than once to convey it properly. You may also wish to draw individual transition states on separate sticky notes (see “Storyboarding iPad Transitions,” Boxes and Arrows, January 5, 2011, http://www.designcaffeine.com/articles/storyboarding-ipad-mobile-transitions/).
* Keyboards are easily mocked up by having another smaller sticky note that can be overlaid on top of the sticky-note pack “device” the study participant is holding. This way any screen can be converted into the “keyboard entry screen” dynamically at any time during the test. This technique gives additional flexibility to the prototype while removing the chore of drawing a complex keyboard design element over and over again.
* Remember to revisit your storyboard. Your wireframe’s workflow should follow your product vision. For example, compare Figure 4-1 with Figure 4-3; you can see that the wireframes are a natural extension of the original vision with a few more details and interface elements added. For example, shortly after I started the testing, it became apparent that people sometimes wanted to take a bunch of pictures first and sell items later. Thus, I added another screen that gave the customer a choice: Take a Picture or Choose Existing (picture) (refer to Figure 4-3). These kinds of changes are common. They are the reason you want to test your idea on paper in the first place! If the change is drastic enough, you must update your vision storyboard accordingly after you learn new insights during your RITE study. In this case, the additional page did not change the basic storyboard scenario—it merely added fidelity to it—so there was no need to change the original vision storyboard (refer to Figure 4-1).

Sticky-note prototypes enable you to quickly and inexpensively explore multiple design approaches while dispensing with elaborate camera equipment and other gadgets. Sticky-note prototypes also enable you and your entire team to escape the confines of the office and boldly go to where the mobile interaction is actually taking place: coffee shops, busy street corners, taxis, and subways. There is tremendous value in putting your paper prototypes in the hands of potential customers in situ, on location where the interaction for which you are trying to design is actually taking place.

At the same time, study participants can be comfortable brainstorming valuable ideas because the prototypes do not have that “finished” look. This leads to powerful collaborative “jam” design sessions with the customers that take place directly in the context where they would be using your creation. These sessions yield invaluable insights you directly incorporate into your designs, much faster than you ever thought possible. That is why I included hand-drawn sticky-note wireframes with nearly every design pattern in the book. Hopefully, these will give you an inspiration to build your own sticky-note prototypes and test them in context, directly with your target customers. If you need help, please visit the companion site for this book (http://AndroidDesignBook.com) where I host videos of example hands-on RITE studies using sticky-note prototypes, mobile use case storyboards and many other resources designed to help you get the most out of this powerful technique.

#### Step 4: Visual Design

A RITE study is not the final deliverable; it’s merely a key step in the design process. The best way to think about your design process is that the state of your prototype and deliverables should reflect the overall state of completion of your product. That’s why early in the process, you will be able to move quickly by employing a lightweight design process to focus on designing the rough flows and screen layouts that work for the customer. After the rough flows have been worked out and tested, it’s time to move on to the final step: visual design.

It is customary to employ visual designers and content managers to give the final project that showroom luster. Although the visual design is not the focus of this book, Chapters 1 and 2 include a few essential pointers on the subject.

It is worth noting that visual design can both enhance and detract from the interaction design intent. Sometimes visual design can make a big difference. Thus it is a good idea to test the app a few more times to ensure that, despite the evolution of the design, the final version remains true to the original vision storyboard’s simplicity and elegance. Styling can be key to creating (or destroying) emotional connections, so it can be worth testing for that as well.

To run the final tests, simply hand the testing device (careful now!) to the person whose opinion you are interested in, perhaps while waiting in line at a coffee shop. Say, “Let me know what you think about the app, and I’ll buy your morning coffee.” This final testing of five to eight people should take no longer than an hour or so. This is an excellent test: Your final flow should ideally take no longer than it does for a person to reach the coffee counter during the morning rush, and the interface should be engaging and self-explanatory enough for the person to manipulate while in the decaffeinated state of mind.