

## 第11章 项目实施部署

# 第11章 项目实施部署

## 章节目标

- 1. 了解项目实施的流程；
- 2. 掌握针对不同对象展现解决方案的方法；
- 3. 掌握解决方案设计师在项目实施阶段的工作内容；
- 4. 了解软件测试的步骤和环节；
- 5. 了解用户体验的概念和它的设计过程。

# 本章目录

- 11.1 项目实施流程
- 11.2 解决方案的展示与提交
- 11.3 具体协助项目实施和部署
- 11.4 软件测试简介
- 11.5 用户体验

## 11.1 项目实施流程

- 通常来讲一个电子商务解决方案系统开发与实施分以下步骤：
  - 收集和评估客户商务需求
  - 对商务需求的当前进度和基础设施进行识别并记录
  - 设计按需应变的商务解决方案
  - 确认这个按需应变的商务解决方案设计
  - 将解决方案展示给客户

- 以上均为解决方案设计师的任务。当客户采纳提交的解决方案后，就由项目管理小组和设计开发人员来接手后续任务了。一般要进行详细设计，就是把解法具体化，也就是回答“应该怎样具体地实现这个系统”这个关键问题。这个阶段的任务还不是编写程序，而是设计出程序的详细规格说明。接下来进行编码和测试，这个阶段的关键任务是写出正确的容易理解、容易维护的程序模块。并通过各种类型的测试(及相应的调试)使软件达到预定的要求。
- 最后将通过各种测试的电子商务应用系统在客户企业进行发布、部署、使用。最后在具体使用过程中肯定或出现这样或那样的问题，因此还必须向客户提供软件维护服务。

- 另外解决方案设计师还在预销售人员和项目实施团队间起一个桥梁作用，这是因为解决方案设计师通常具有客户业务分析、跨产品的技术知识、按需应变电子商务知识、丰富的设计、展示和提交电子商务解决方案的经验。所以在项目销售、实施和部署阶段，解决方案设计师经常和销售经理、IT服务经理在一起工作。

## 11.2 解决方案的展示与提交

- 协助销售团队来准备解决方案的展示。主要包括：
- 将设计中考虑的事项与提议的评价标准作映射，用来确保提案能达到甚至超过客户的接受标准；
- 从设计角度对客户各种问题以及客户感兴趣的内容提供专家意见；
- 同时能灵活的提供一些替代解决方案来满足用户特殊需求；
- 协助证明提交的按需应变电子商务解决方案的商业价值

## 11.2.1 根据客户对象的不同对展示内容作裁减

- 为了让客户能采纳提议的解决方案，解决方案向客户的展示是第一步，也是最重要的一步。这是由于真正的按需应变的电子商务必须能满足非常广泛的各个层面的商业需求。不同的客户对象由于各自的职责不同，他们对按需应变的电子商务的需求肯定也不尽相同。向客户展示（或协助展示）解决方案的解决方案设计师必须是一个给人深刻印象的展示者。他同时也必须是一个好的听众。并且要能就客户特定的商业、技术知识以及客户面临的关键问题进行轻松的讨论。
- 解决方案方案设计师在展示之前提前了解展示对象的组成非常关键。这能使解决方案设计师能将展示个性化，使得展示能适合客户的期望。通俗来讲就是“见什么人说什么话”。展示快结束时，发布展示资料、宣传册之前，花点时间来收集客户反馈并确定下一步内容是一个好的主意。



## 11.2.2采用适当的表达方式来展示解决方案

- 解决方案设计师通常可以采用解决方案通览(solution walk through)或原型法来验证设计需求。
- 解决方案通览一般通过将展示活动和宣传资料结合起来完成。
- 原型系统通常用于销售过程。当客户对你是否如你所宣称的那样有能力实现他们的需求产生疑问时，此时，一个原型系统往往可以显示解决方案的确有能力满足他们的商业需求。
- 由于原型系统通常在销售阶段演示给客户看，所以它通常比较简单，一般是完全基于客户的商业需求的。有时候原型系统不能完全满足客户需求，这时可以将它发展成为一个解决方案演练，以满足客户需求。但在整个项目过程中，为了验证解决方案满足用户需求，在不断重复的设计过程中，原型系统通常会得到改良。这时，可以将它发展成为一个解决方案演练，以满足客户需求。

## 11.2.2 采用适当的表达方式来展示解决方案

- 在将一个成熟的解决方案应用于一个产品环境中前，须使用解决方案演练，它提供给客户最后一次机会提出反馈意见。它不仅用来接受客户反馈，确保解决方案设计的正确执行，而且还是一种销售工具，以建立与客户的进一步合作机会。解决方案演练在后面的项目实施过程中也会使用。通常在系统测试时使用它做测试的导航。

## 11.2.3 解决方案设计中对RIO和TCO的考虑

- 前面章节已经讨论过了ROI和TCO目标。此时解决方案设计师就要将方案中对ROI和TCO的各种具体考虑展示给客户看，让客户相信ROI和TCO目标靠你的解决方案能达到。

## 11.2.4 针对客户对设计的反馈信息作陈述

- 解决方案设计师必须能回应客户提出的各种与设计相关的问题和有意义的评论，一定要让客户满意。要做到这一点，解决方案设计师可能需要回到前面任何设计步骤来澄清商业需求，如果当初需求分析或设计有误的话还要对解决方案进行修改。如果变动比较大的话还需要对整个解决方案进行重新验证。另外还因该举行一个会议来确认客户的反馈信息已经整合到解决方案中。在最终解决方案正式实施前必须得到客户的正式批准。

## 11.3 具体协助项目实施和部署

- 当设计的解决方案被客户批准，项目就转由项目经理来接管了。一般解决方案设计师会以为他们的任务和责任结束了。其实不然，在整个项目的实施和部署的所有阶段对解决方案设计师的技能时不时有需要。例如，解决方案设计师应该帮助项目管理团队来：
  - 对项目完成所需时间和资源（人力资源和技术资源）进行大致评估。
  - 说明设计中对项目实施存在的潜在风险。
  - 帮助创建客户接受标准和项目终点。
  - 检查工作声明 (statement of work)，确保它符合技术说明书。
  - 在实施过程中，当有人需要时提供技术指南
  - 检查可接受性测试结果。
  - 验证按需应变电子商务解决方案达到客户目标。

## 11.3 具体协助项目实施和部署

- 解决方案设计师因该知道上面所有的各类信息都已经在解决方案中作了记录，均应有文档可查的。这些内容应该包括：
  - 商业目标 参看前面章节相关内容
  - 交流计划：计划中应一一指出所有直接或间接与项目相关的各方，并说明由谁对项目实施全方位负责。这个计划也应该描述万一出现问题所应采取的处理流程。具体来讲，就像一个很简单的列表，列表内容为姓名、电话号码和对项目的责任。
  - 培训计划：对成功实施项目所需要的技能、客户当前的技能作出评估。并估计填补技术缺陷的步骤。
  - 技术支持计划：这个计划主要涉及在项目实施过程中客户将从谁、收到的什么样的技术支持。



## 11.3 具体协助项目实施和部署

- 备份和恢复计划：确定备份和恢复的必要性，备份和恢复怎样影响客户的内部服务级别的协定，并确定谁负责备份和恢复。
- 系统管理计划：这个计划具体描述客户将怎样部署、维护和跟踪软件。它也会涉及这样一些项目：配置、运作、性能、可用性、存储和安全。
- 首次展示和进度表：用一个进度表j将前面列出的所有单独计划按照先后顺序包括进来。把这些活动和它们的进度表摆出来有助于客户直观的看到并理解项目的时间线以及项目实施的依赖性。
- 服务级协议：提供系统的可用性、扩充性、伸缩性、性能和安全性保障的协议。
- 依赖性：陈述对解决方案成功实施很关键的人力和其他资源的可用性的设想。

## 11.3 具体协助项目实施和部署

- 备份和恢复计划：确定备份和恢复的必要性，备份和恢复怎样影响客户的内部服务级别的协定，并确定谁负责备份和恢复。
- 系统管理计划：这个计划具体描述客户将怎样部署、维护和跟踪软件。它也会涉及这样一些项目：配置、运作、性能、可用性、存储和安全。
- 首次展示和进度表：用一个进度表j将前面列出的所有单独计划按照先后顺序包括进来。把这些活动和它们的进度表摆出来有助于客户直观的看到并理解项目的时间线以及项目实施的依赖性。
- 服务级协议：提供系统的可用性、扩充性、伸缩性、性能和安全性保障的协议。
- 依赖性：陈述对解决方案成功实施很关键的人力和其他资源的可用性的设想。



## 11.3.1 预定相关服务和资源来部署解决方案

- 解决方案设计师可能会时不时对解决方案实施/部署团队就服务和资源提出建议。例如在项目实施阶段，实施团队可能会碰到建议的性能问题。解决方案设计师能帮助实施团队通过提供相关的技术资料或把他们委托给某个专家或特殊IT服务商来解决这些问题。解决方案设计师也能就实施团队所缺乏相关技能来处理的特定方面提供指导。如，网络方面、安全或隐私方面等等。

## 11.3.2 为项目实施中的电话求助和会议提供指导

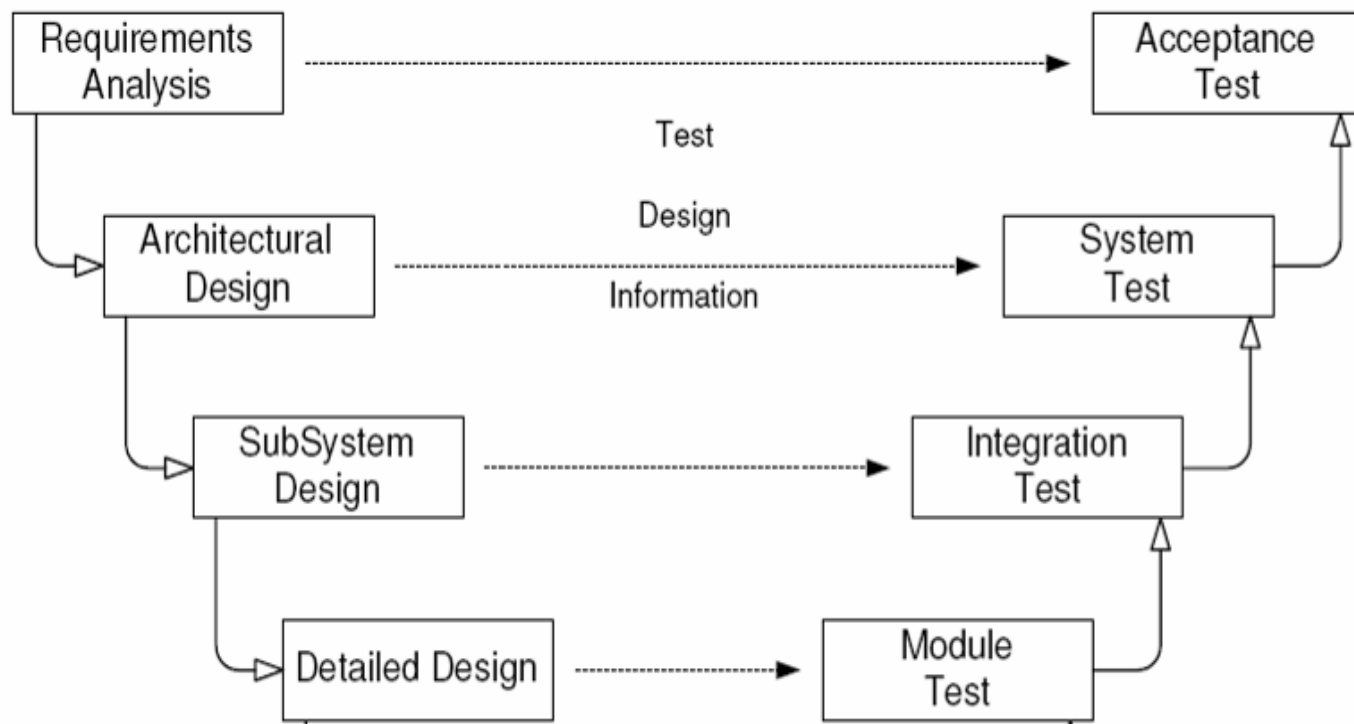
- 解决方案设计师应该对客户的业务、被实施的解决方案、解决方案要满足的商业和技术需求等方面有一个大的总体视图。它们在解决方案实施过程中的价值是提醒实施者那些解决方案必须解决的关键业务问题，提醒解决方案设计作出的决定后面的理由，提醒最关键的设计因素和衡量解决方案成功的标志。通常对这些事项的高级视图会在解决方案的具体实施中丢失。因此解决方案设计师能保持项目实施团队关注这些问题而不会偏离重要的目标。

### 11.3.3 为成本、时间和资源的估算提供输入信息

- 一个解决方案设计师能在成本、时间和资源方面为项目管理团队提供帮助。解决方案设计师处于一种独一无二的位置来提供这些帮助。因为他/她最了解解决方案的技术架构，商业需求和为项目考虑的各种选项。

## 11.4 软件测试简介

- 软件测试一般先指定详细的测试计划，然后由测试人员进行各种测试，最后完成测试报告。软件测试按照测试顺序一般有以下种类：
- 1、单元测试
- 2、集成测试
- 3、确认测试
- 4、系统测试



# 1、单元测试

- 单元测试又称为模块测试，是针对程序模块（软件设计的最小单位）来进行正确性检验的测试工作。单元测试的目的是检测程序模块对《详细设计说明书》的符合程度；软件单元测试依据是《单元测试计划》。单元测试由测试工程师编制测试用例进行测试，及针对程序模块进行多次循环反复的单元测试，并将测试结果记录在针对单元测试的《软件测试报告》上。若程序模块通过单元测试，则按《配置管理规范》所规定的标识方法进行标识。
- 单元测试的内容主要包括：
  - 模块接口测试
  - 局部数据结构测试
  - 路径测试
  - 错误处理测试
  - 边界测试

## 2、软件集成测试

- 软件集成测试又称组装测试，即对程序模块采用自顶向下或自底向上组装起来，对系统的接口进行正确性检验的测试工作。软件集成测试由项目经理组织软件测试工程师依据《概要设计说明书》和《集成测试计划》进行。测试人员应提交针对软件集成测试的《软件测试报告》，项目经理负责对软件集成测试结果的进行确认。通过集成测试，则按《配置管理规范》所规定的标识方法进行标识。集成测试通常是在单元测试的基础上，需要将所有模块按照设计要求组装成为系统。这时需要考虑的问题是：
  - 在把各个模块连接起来的时候，穿越模块接口的数据是否会丢失；
  - 一个模块的功能是否会对另一个模块的功能产生不利的影响；
  - 各个子功能组合起来，能否达到预期要求的父功能；
  - 全局数据结构是否有问题；
  - 单个模块的误差累积起来，是否会放大，从而达到不能接受的程序。



### 3、确认测试

- 确认测试又称验收测试。确认测试的任务是验证软件的功能和性能及其他特性是否与用户的要求一致。对软件的功能和性能要求在软件需求规格说明中已经明确规定。
- 确认测试中的有效性测试。有效性测试是在模拟的环境下，运用黑盒测试的方法，验证所测软件是否满足需求规格说明书中所列的需求。
- 在全部软件测试的测试用例运行完后，所有的测试结果可以分为两类：
  - 测试结果与预期的结果相符；
  - 测试结果与预期的结果不符。

## 4、系统测试

- 系统测试是将通过确认测试的软件，作为整个基于计算机系统的的一个元素，与计算机硬件、外设、某些支持软件、数据和人员等其他系统元素结合在一起测试。在实际运行（使用）环境下，对计算机系统进行一系列的集成测试和确认测试。
- 系统测试的目的在于通过与系统的需求定义作比较，发现软件与系统定义不符合或与之矛盾的地方。



## 系统测试的类型：

- 恢复测试：恢复测试是要证实在克服硬件故障后，系统能否正常地继续进行工作，并不对系统造成任何损害。为此，可采用各种人工干预的手段，模拟硬件故障，故意造成软件出错。
- 安全性测试：系统的安全性测试是要检验在系统中已经存在的系统安全性、保密性措施是否发挥作用，有无漏洞。强度/压力(stress)测试：让程序/应用在高负荷情况下运行。
- 敏感性测试：在程序有效数据界限内一个非常小的范围内的一组数据可能引起极端的或不平稳的错误处理出现，或使性能极度下降。
- 性能测试：性能测试是要检查系统是否满足在需求说明书中规定的性能。

## 系统测试的类型：

- 功能测试：目标中的功能是否真正实现了。
- 批量测试：证明程序不能处理目标中的大批数据。
- 存储量测试：测试程序所占用的内外存容量（静/动态）。
- 配置测试：至少每一类和最大最小的设备配置情况都要测试。
- 兼容/移植测试：对现有程序进行修改和补充后，要进行此类测试。
- 可安装性测试：测试系统的安装过程。
- 可靠性测试：如平均无故障时间。
- 可维护性测试：对维护过程和难易程度进行测试。
- 文档测试：审查文档的正确性，对文档中的每个例子都要作为测试用例。
- 工序测试：测试操作工序的次序正确性

## 11.5 用户体验

- 在项目的设计、部署、实施过程中，都会涉及到用户体验。这里对它做一个简单的介绍。

## 11.5.1 什么是用户体验

- 用户体验，英文叫做user experience，缩写为UE，或者UX。
- 英文原文如下：the user experience, mostly called "customer experience" when referring to e-commerce websites; the totality of the experience of a user when visiting a website. Their impressions and feelings. Whether they're successful. Whether they enjoy themselves. Whether they feel like coming back again. The extent to which they encounter problems, confusions, and bugs.

## 11.5.1 什么是用户体验

- 用户体验主要是来自用户和人机界面的交互过程。在早期的软件设计过程中，人机界面被看做仅仅是一层包裹于功能核心之外的“包装”而没有得到足够的重视。其结果就是对人机界面的开发是独立于功能核心的开发，而且往往是在整个开发过程的尾声部分才开始的。这种方式极大地限制了对人机交互的设计，其结果带有很大的风险性。因为在最后阶段再修改功能核心的设计代价巨大，牺牲人机交互界面便是唯一的出路。这种带有猜测性和赌博性的开发几乎是难以获得令人满意的用户体验。至于客户服务，从广义上说也是用户体验的一部分，因为它是同产品自身的设计分不开的。客户服务更多的是对人员素质的要求，而已经难以改变已经完成并投入市场的产品了。但是一个好的设计可以减少用户对客户服务的需要，从而减少公司在客户服务方面的投入，也降低由于客户服务质量引发用户流失的机率。

## 11.5.2 用户体验的设计过程(design process)

- 现在流行的设计过程注重以用户为中心（**user centered**）。用户体验的概念从开发的最早期就开始进入整个流程，并贯穿始终。其目的就是保证（1）对用户体验有正确的预估（2）认识用户的真实期望和目的（3）在功能核心还能够以低廉成本加以修改的时候对设计进行修正（4）保证功能核心同人机界面之间的协调工作，减少**BUG**。
- 在具体的实施上，就包括了早期的**focus group**（焦点小组），**contextual interview**，和开发过程中的多次**usability study**（可用性实验），以及后期的**user test**（用户测试）。在设计--测试--修改这个反复循环的开发流程中，可用性实验为何时出离该循环提供了可量化的指标。



## 11.5.3 用户体验的评估过程

- IBM公司对用户体验有以下评估过程。详情可看  
[http://www-3.ibm.com/ibm/easy/eou\\_ext.nsf/Publish/2830](http://www-3.ibm.com/ibm/easy/eou_ext.nsf/Publish/2830)
- 用户体验的测试不仅仅是对设计和UI的应用，也可以在功能测试中去应用。功能测试中用户体验测试倾向于一种基础的集成测试。就是用户需要将多个功能或者服务综合起来去运行，达到一种体验，同时也保证了在功能测试和单元测试中可能漏掉的地方。这不是简单的集成测试，因为在这种集成中一定要本着用户的角色和感受去保证质量，而用户也不是简单的个性化用户，是要通过相当多的商务调查和综合最后得出的你的产品的各个典型用户，这才算得上真正的用户体验。测试人员就是要投入到典型角色中去体验的。