**软件工程架构设计**

一、需求分析

1.相关[系统分析员](https://baike.baidu.com/item/%E7%B3%BB%E7%BB%9F%E5%88%86%E6%9E%90%E5%91%98" \t "_blank)向用户初步了解需求，然后用相关的工具软件列出要开发的系统的大功能模块，每个大功能模块有哪些小功能模块，对于有些需求比较明确相关的界面时，在这一步里面可以初步定义好少量的界面。

2.系统分析员深入了解和分析需求，根据自己的经验和需求用WORD或相关的工具再做出一份文档系统的功能需求文档。这次的文档会清楚列出系统大致的大功能模块，大功能模块有哪些小功能模块，并且还列出相关的界面和界面功能。

3.系统分析员向用户再次确认需求。

二、概要设计

首先，开发者需要对软件系统进行[概要设计](https://baike.baidu.com/item/%E6%A6%82%E8%A6%81%E8%AE%BE%E8%AE%A1" \t "_blank)，即[系统设计](https://baike.baidu.com/item/%E7%B3%BB%E7%BB%9F%E8%AE%BE%E8%AE%A1)。概要设计需要对软件系统的设计进行考虑，包括系统的[基本处理](https://baike.baidu.com/item/%E5%9F%BA%E6%9C%AC%E5%A4%84%E7%90%86)流程、系统的组织结构、模块划分、功能分配、接口设计、运行设计、数据结构设计和出错处理设计等，为软件的[详细设计](https://baike.baidu.com/item/%E8%AF%A6%E7%BB%86%E8%AE%BE%E8%AE%A1)提供基础。

三、详细设计

在[概要设计](https://baike.baidu.com/item/%E6%A6%82%E8%A6%81%E8%AE%BE%E8%AE%A1" \t "_blank)的基础上，开发者需要进行软件系统的详细设计。在[详细设计](https://baike.baidu.com/item/%E8%AF%A6%E7%BB%86%E8%AE%BE%E8%AE%A1)中，描述实现具体模块所涉及到的主要算法、数据结构、类的[层次结构](https://baike.baidu.com/item/%E5%B1%82%E6%AC%A1%E7%BB%93%E6%9E%84)及调用关系，需要说明软件系统各个层次中的每一个程序(每个模块或[子程序](https://baike.baidu.com/item/%E5%AD%90%E7%A8%8B%E5%BA%8F" \t "_blank))的设计考虑，以便进行编码和测试。应当保证软件的需求完全分配给整个软件。详细设计应当足够详细，能够根据详细设计报告进行编码。

拓展知识点介绍：

1. 需求分析设计工具：StarUML
2. StarUML: StarUML(简称SU)，是一种创建UML类图，生成类图和其他类型的[统一建模语言](https://baike.baidu.com/item/%E7%BB%9F%E4%B8%80%E5%BB%BA%E6%A8%A1%E8%AF%AD%E8%A8%80)(UML)图表的工具。StarUML是一个开源软件。
3. UML：Unified Modeling Language (UML)又称统一建模语言或[标准建模语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E5%BB%BA%E6%A8%A1%E8%AF%AD%E8%A8%80)，是始于1997年一个[OMG](https://baike.baidu.com/item/OMG/3041465)标准，它是一个支持模型化和软件系统开发的图形化语言，为软件开发的所有阶段提供模型化和可视化支持，包括由需求分析到规格，到构造和配置。 面向对象的分析与设计(OOA&D，OOAD)方法的发展在80年代末至90年代中出现了一个高潮，[UML](https://baike.baidu.com/item/UML)是这个高潮的产物。它不仅统一了Booch、Rumbaugh和Jacobson的表示方法，而且对其作了进一步的发展，并最终统一为大众所接受的[标准建模语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E5%BB%BA%E6%A8%A1%E8%AF%AD%E8%A8%80)。
4. 用例图：

用例图是指由参与者（Actor）、[用例](https://baike.baidu.com/item/%E7%94%A8%E4%BE%8B)（Use Case），边界以及它们之间的关系构成的用于描述系统功能的视图。用例图（User Case）是外部用户（被称为参与者）所能观察到的系统功能的模型图。用例图是系统的蓝图。用例图呈现了一些参与者，一些用例，以及它们之间的关系，主要用于对系统、子系统或类的功能行为进行[建模](https://baike.baidu.com/item/%E5%BB%BA%E6%A8%A1)。

1. 类图：

类图(Class diagram)由许多（静态）说明性的模型元素（例如类、包和它们之间的关系，这些元素和它们的内容互相连接）组成。类图可以组织在（并且属于）包中，仅显示特定包中的相关内容。

类图(Class diagram)是最常用的[UML图](https://baike.baidu.com/item/UML%E5%9B%BE)，显示出类、接口以及它们之间的静态结构和关系；它用于描述系统的结构化设计。

类图(Class diagram)最基本的元素是类或者接口。

1. E-R图：

E-R图也称实体-联系图(Entity Relationship Diagram)，提供了表示实体类型、属性和联系的方法，用来描述现实世界的[概念模型](https://baike.baidu.com/item/%E6%A6%82%E5%BF%B5%E6%A8%A1%E5%9E%8B/3187025)。

它是描述现实世界关系概念模型的有效方法。是表示概念关系模型的一种方式。用“矩形框”表示实体型，矩形框内写明实体名称；用“椭圆图框”表示实体的属性，并用“实心线段”将其与相应关系的“实体型”连接起来；

用”菱形框“表示实体型之间的联系成因，在[菱形](https://baike.baidu.com/item/%E8%8F%B1%E5%BD%A2)框内写明联系名，并用”实心线段“分别与有关实体型连接起来，同时在”实心线段“旁标上联系的类型（1:1,1:n或m:n）。

实体联系模型，实体[关系模型](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E6%A8%A1%E5%9E%8B)或实体联系模式图(ERD)是由美籍华裔计算机科学家陈品山(Peter Chen)发明，是概念数据模型的高层描述所使用的数据模型或模式图，它为表述这种实体联系模式图形式的[数据模型](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E6%A8%A1%E5%9E%8B)提供了图形符号。这种数据模型典型的用在[信息系统](https://baike.baidu.com/item/%E4%BF%A1%E6%81%AF%E7%B3%BB%E7%BB%9F)设计的第一阶段；比如它们在[需求分析](https://baike.baidu.com/item/%E9%9C%80%E6%B1%82%E5%88%86%E6%9E%90)阶段用来描述信息需求和/或要存储在数据库中的信息的类型。但是数据建模技术可以用来描述特定论域(就是感兴趣的区域)的任何本体(就是对使用的术语和它们的联系的概述和分类)。在基于数据库的信息系统设计的情况下，在后面的阶段(通常叫做逻辑设计)，[概念模型](https://baike.baidu.com/item/%E6%A6%82%E5%BF%B5%E6%A8%A1%E5%9E%8B)要映射到逻辑模型如[关系模型](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E6%A8%A1%E5%9E%8B)上；它依次要在物理设计期间映射到物理模型上。注意，有时这两个阶段被一起称为"物理设计"。

四、编码

在[软件编码](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E7%BC%96%E7%A0%81)阶段，开发者根据《软件系统详细设计报告》中对数据结构、算法分析和模块实现等方面的设计要求，开始具体的编写程序工作，分别实现各模块的功能，从而实现对目标系统的功能、性能、接口、界面等方面的要求。在规范化的研发流程中，编码工作在整个项目流程里最多不会超过1/2，通常在1/3的时间，所谓磨刀不误砍柴功，设计过程完成的好，编码效率就会极大提高，编码时不同模块之间的进度协调和协作是最需要小心的，也许一个小模块的问题就可能影响了整体进度，让很多程序员因此被迫停下工作等待，这种问题在很多研发过程中都出现过。编码时的相互沟通和应急的解决手段都是相当重要的，对于程序员而言，bug永远存在，你必须永远面对这个问题，大名鼎鼎的微软，可曾有连续三个月不发补丁的时候吗？从来没有！

五、测试

测试编写好的系统。交给用户使用，用户使用后一个一个的确认每个功能。软件测试有很多种：按照测试执行方，可以分为内部测试和外部测试；按照测试范围，可以分为模块测试和整体联调；按照测试条件，可以分为正常操作情况测试和异常情况测试；按照测试的输入范围，可以分为全覆盖测试和抽样测试。以上都很好理解，不再解释。总之，测试同样是项目研发中一个相当重要的步骤，对于一个大型软件，3个月到1年的外部测试都是正常的，因为永远都会有不可预料的问题存在。完成测试后，完成验收并完成最后的一些帮助文档，整体项目才算告一段落，当然日后少不了升级，修补等等工作，只要不是想通过一锤子买卖骗钱，就要不停的跟踪软件的运营状况并持续修补升级，直到这个软件被彻底淘汰为止。

六、软件交付

在软件测试证明软件达到要求后，软件开发者应向用户提交开发的目标安装程序、数据库的[数据字典](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%AD%97%E5%85%B8)、《用户安装手册》、《用户使用指南》、需求报告、设计报告、[测试报告](https://baike.baidu.com/item/%E6%B5%8B%E8%AF%95%E6%8A%A5%E5%91%8A)等双方合同约定的产物。

《用户安装手册》应详细介绍安装软件对运行环境的要求、安装软件的定义和内容、在客户端、服务器端及[中间件](https://baike.baidu.com/item/%E4%B8%AD%E9%97%B4%E4%BB%B6)的具体安装步骤、安装后的系统配置。

《用户使用指南》应包括软件各项功能的使用流程、操作步骤、相应业务介绍、特殊提示和注意事项等方面的内容，在需要时还应举例说明。