



1.1 软件文档的意义



人们在开始做一件事情时，首先应弄清楚为什么要做这件事情，并由事情本身的重要性、意义来决定是做还是不做，只有明白为什么要做之后，才会心甘情愿地去做。有做的意愿之后，才是如何做的问题。

软件对于客户来说，他们能接触到的就是软件开始运行及运行之后的结果，至于软件是如何实现客户要求的，这是客户无法知道且没有必要知道的。但是，如何使用该软件，软件开发方一般是难以亲自指导客户的。真正让客户学会使用软件的有效方法是，让客户参照使用说明书，即用户手册。从软件开发方来看，由于现在软件系统的规模越来越大，一般已经不是一个人或几个人的小团队能够独立完成的系统，随着系统复杂程度的不同，开发人数从一人到几千人。人数增加到一定程度时，沟通所带来的时空成本都将成指数倍地增长，且最终将不可收缩。此外，今天的软件系统的开发通常不是由单纯的一个团队来完成的，一个大系统往往被分成若干小系统，由不同的开发团队来完成。从时间上来看，一个大系统可能由某个团队承担设计工作，而由另一个团队承担编码，再由另一个团队完成测试工作。系统如何分割，如何把子系统的要求表达清楚，这些问题如果只是依赖简单的口头说明，也许在开始的一段时间内是可以记住的，但是随着时间的增长，记忆开始遗忘，这样的开发模式更增加了开发者之间的沟通难度。

软件文档(document)描述系统功能，刻画子系统间的相互关系，是为开发者提供的精确、完整的指导资料。软件文档是软件开发开发者之间的沟通渠道，是具体工作的安排表，是系统的开发标准。

软件文档是软件产品的伴生物，记录着软件产品从诞生之前到开发完成整个过程的相关信息，它具有固定不变的形式，可被人和计算机阅读。它和计算机程序共同构成了能完成特定功能的计算机软件。传统的硬件产品及其产品资料在整个生产过程中都是有形可见的，但是软件生产则有很大不同，软件文档本身就是软件产品的一部分。没有软件文档的软件，不能称其为软件，更谈不上软件产品。没有软件文档的软件是不利于推广、不可维护、无法重用的。软件文档的编制(documentation)在软件开发工作中占有突出的地位和相当的工作量。高效率、高质量地开发、分发、管理和维护软件文档，对于转让、变更、修

正、扩充和使用软件文档，对于充分发挥软件产品的效益有着重要意义。

然而，在实际工作中，软件文档在编制和使用中存在着许多有待于解决的问题。软件开发人员中较普遍地存在着对编制文档不感兴趣的现象。从用户方面看，他们又常常抱怨软件文档售价太高、软件文档不够完整、软件文档已经陈旧或软件文档太多、难于使用等。

许多软件文档存在以下问题：

- 错误的语法和/或拼错的词语
- 不完整
- 过时或不准确
- 过于冗长
- 未经解释的缩略语或专用术语
- 查找信息困难

存在这些问题的主要原因是软件文档经常被我们放到次位，因为工程预算迫使我们优先考虑开发过程中的主要活动，也就是那些可以看得到利润的地方。编写文档需要成本，因而它经常成为一项主观上的活动，而且通常被认为没有重要作用，应该尽量避免。许多项目经理认为客户不需要文档，它只是用来装点门面的。

软件文档质量差的另一个原因在于文档撰写者。许多应用程序开发经理认为软件文档的编写是软件开发过程的一个标准组成部分，因此要求开发人员在编码的过程中产出文档。

尽管这种做法在理论上行得通，但它未考虑开发人员编写文档的能力。简单来说，技术人员是用来开发软件而不是编写文档的。为了解决这个问题，许多应用程序开发经理雇用专业技术文档编写者或业务分析师，以期改进软件文档的质量。但这又碰到了另一个难题：专业编写者及业务分析师的技术水平有限。

解决这个问题要考虑需要编写的文档以及文档的预期读者。一般的规则是，写文档需要团队协作，这样就需要开发人员和文档编写者利用彼此的优点，取长补短。例如，假如预期读者是系统设计师，那么开发人员需要提供技术细节，然后文档编写者按照正确语法组织和编辑内容。

1.2 软件文档的作用



只有真正明白软件文档在软件整个生命周期中的作用之后，才能积极并认真地付出时间和精力去撰写软件文档。

软件文档的本质作用是桥梁、是纽带，连接着软件开发方、管理人员、用户及计算机，将其构成一个相互影响、相互作用的整体。软件开发人员在各个阶段中以软件文档作为前阶段工作成果的体现和后阶段工作的依据，这个作用是显而易见的。软件开发过程中软件

开发人员需制定一些工作计划或工作报告，这些计划和报告都要提供给管理人员，并得到必要的支持。管理人员则可通过这些软件文档了解软件开发项目安排、进度、资源使用和成果等。软件开发人员需为用户了解软件的使用、操作和维护提供详细的资料，这被称为用户文档。

在软件工程中，文档用来表示对需求、工程或结果进行描述、定义、规定、报告或认证的任何文字或图示的信息。它们描述和规定了软件设计和实现的细节，说明使用软件的操作命令。文档也是软件产品的一部分，没有文档的软件就不成为软件。软件文档的编制在软件开发过程中占有突出的地位和相当大的工作量。高质量的文档对于转让、变更、修改、扩充和使用文档，对于发挥软件产品的效益有着重要的意义。

软件文档的最主要目标是传达一个系统的技术要素和使用方法。第二个目标是提供软件开发过程中的需求、决策、行为、角色和责任的书面记录。只有实现了这两个目标，软件文档才真正提供了有意义的信息。

软件开发人员在各个阶段中以文档作为前阶段工作成果的体现和后阶段工作的依据，这个作用是显而易见的。软件开发过程中软件开发人员需制定一些工作计划或工作报告，这些计划和报告都要提供给管理人员，并得到必要的支持。管理人员则可通过这些文档了解软件开发项目的安排、进度、资源使用和成果等。软件开发人员需为用户了解软件的使用、操作和维护提供详细的资料，我们称此为用户文档。以上三种文档构成了软件文档的主要部分。

1.3 软件文档的分类



1.3.1 可行性研究报告

可行性研究报告的编写目的是说明该软件开发项目的实现在技术、经济和社会条件方面的可行性，评述为了合理地达到开发目标而可能选择的各种方案，说明并论证所选定的方案。可行性研究报告的内容包括可行性研究的前提、对现有系统的分析、所建议的系统、可选择的其他系统方案、投资及效益分析、社会因素方面的可行性和结论。

1.3.2 项目建议书

项目建议书为软件项目实施方案制订出具体计划，包括市场分析，项目的概要介绍，项目的赢利模式，项目的整体框架，各部分工作的负责人员、开发进度、开发经费的开发预算、所需的硬件及软件资源等。项目建议书一般由项目经理根据客户的开发计划来编写，作为整个项目的整体规划，未来的开发工作都基于这个计划来执行。

进行可行性分析是一个自我否定的过程，而写项目建议书是一个向别人阐述自己观点

的过程。而且项目建议书一般情况下是要去说服你的上司或者投资人来做这个项目，所以一定要非常完善，把所有可能的利弊都分析到。

1.3.3 招标投标文件

国内的软件项目招标文件的写作规则并不存在行业标准。一般的招标文件内容包括项目招标简介、企业信息化项目需求、咨询与实施需求、售后服务要求和信息系统要求等。投标文件内容包括投标人商务文件构成、投标文件要求和项目建议书的写作要求等。

1.3.4 需求分析书

需求分析书由软件开发人员与客户共同编写，客户用自然语言描述对系统的功能和性能方面的预期效果，开发人员将客户需求转化为文字记录下来，尽可能充分地诱导出客户的需求，使未来开发出来的系统能够真正满足客户的需要，与客户预期的系统相差无几。需求分析书是面向客户的软件文档，包括产品概述、主要概念、操作流程、功能列表和解说、注意事项、系统环境等。对系统进行详细的功能分析(包括客户提出的要求和根据开发经验建议的功能)，描述本产品是什么，有什么特殊的概念，包括哪些功能分类，需要具备什么功能，该功能的操作如何，实现时必须注意什么细节，系统运行环境的要求等。

构建一个软件系统，如同建造一所房屋。需求分析书要刻画房屋主人的意图，他想建造一个什么样的屋子，是建一个独门独院的小别墅还是建一个有很多房间的公寓。屋子有几个客厅、几个卧室和几个卫生间，这些房间如何布局等，这些问题都是在需求分析书中必须回答的问题。另一方面，由于房屋主人一般都是建筑方面的外行，就需要房屋设计人员从建筑的角度帮助他们尽可能多地发掘自己的要求。

1.3.5 概要设计书

概要设计书以需求分析书为基础，包括功能实现、模块组成、功能流程图、函数接口、数据字典、软件开发需要考虑的各种问题等。这些问题都是从概要设计书编写开始，就正式进入计算机领域的软件文档了。由于需求分析书随时都有变更的可能性，所以概要设计书制定后也不是一成不变的，它要随着需求分析书的变更而变化，从需求设计书出发，抽象出系统的功能模块，数据库要求，体系结构等大方向的问题。

在建造房屋的这一过程中，概要设计书就相当于房屋主体结构图，此时需要决定的是房屋的格局、用料、管道布局等关键问题，这些问题一旦决定下来，房屋的轮廓就已经出来了。软件的概要设计书也是如此，概要设计书完成之后，软件系统的骨架即可决定，未来的工作就可以在此基础上展开。

1.3.6 详细设计书

概要设计书从高层着手对系统进行描述，但是拿着这样一份概要设计书是无法进行实际编码工作的。详细设计书以概要设计书为基础，对已拆分出来的子系统和功能模块逐个进行设计，这里的设计要详细到每个模块实现的具体步骤，按钮按下完成什么操作，点击某个连接迁移到什么画面，还要绘制出页面原形，提供与数据库的交互方法，数据的表现形式，这些都是要在详细设计书中具体描述的。当然，由于项目复杂度和规模不同，详细设计书的复杂度也会不同，功能简单的系统如果配上复杂的软件文档，只会让软件开发变得更复杂，违背了软件文档建立的目的。相反，如果软件系统复杂度高，参与人员众多，就必须配备详细的软件文档，给不同角色的人员提供尽可能详细而全面的信息。

回到建造房屋这个例子，房屋的骨架搭建起来后，就应丰富它的身体，装修设计如同软件的详细设计，决定屋子的整体风格，屋子四面墙如何安排，是贴墙纸呢还是刷油漆，窗户以及门上的玻璃采用什么花样，灯光如何布置等。装修设计完成之后，未来屋子的全貌就八九不离十了。同样，成功的详细设计书一旦做成，未来软件系统的实现方法也就确定了，编码人员按照详细设计书就可以开始具体的编码工作了。并且由于很多实现细节的问题都在详细设计中考虑分析过，一些技术难题也已经在前期进行攻关试验并最终得出结论，尽量让问题早发现早解决，不至于延迟到项目后期才发现。通过充分的详细设计过程，可以使编码过程变得简单易行，达到事半功倍的效果。

1.3.7 项目验收总结报告

项目验收总结报告的内容包括对所完成系统的测试、验收和总结。测试的目的是为了将软件产品交付给用户之前尽可能多地发现问题并及时修正问题，不至于等到用户在使用过程中才发现问题。测试计划就是规划整个测试的实施过程，计划应包括测试的内容、进度、条件、人员、测试用例的选取原则、测试结果允许的偏差范围等。由于测试会细分为单元测试、集成测试、系统测试等几个环节，测试计划书也应该按照阶段制定。

在软件系统的实际开发过程中，不仅只有以上所述的7种软件文档，还有维护手册、用户手册等。

软件系统开发完毕交付给客户之后，软件的开发周期就结束了。但是从软件的生命周期来看，开发周期只是生命周期的一部分，更多的时间在于软件的服役期。软件与有形产品不同，不会随着时间的推移发生物质损耗与折旧，但是软件在运行过程中会发现这样那样的问题，有可能和客户预想的功能稍有出入，也有可能没有正确实现客户的要求，这就是通常所说的缺陷。几乎没有人敢断言自己开发出来的软件投入使用之后一点问题也没有，区别只是缺陷的多少问题。所以软件在投入使用之后，与有形产品一样存在着维护的问题。

维护工作有可能是由软件系统的开发方来实施，也有可能是客户方自己有相关的技术人员。无论是由开发方还是客户的技术人员进行维护，都需要借助维护手册来实施，因为即使是对于开发方，由于他们在完成一个软件的开发之后，很快就会进入下一个新系统的开发中，并且随着时间的推移，对原来系统的记忆开始模糊，逐渐就不能清楚地回忆起当时开发的细节了。对于客户的维护人员就更难知道如何去维护这个软件系统了。所以，维护手册起到了技术支持的作用，通过查看维护手册能够诊断软件的问题，采取相应的措施给予解决。

为了提供技术支持，维护手册应当包括产品简介、系统须知、初始环境设置、系统配置、数据管理和备份、技术问题解答和联系方式等。由于维护手册面对的是有一定计算机相关知识的人，所以它的内容应该比较专业，力求用严谨的方式刻画问题的解决方法。

用户手册的读者是软件的最终用户，用户需要从手册中获得关于软件系统的各种各样的信息，所以用户手册必须详细地描述软件的功能、性能和用户界面，使用户了解如何使用该软件。用户手册的读者通常都是没有计算机相关知识的人员，这就决定了用户手册书写的语言必须是非专业性质的，与之前提到的概要设计书详细设计书完全不同，它不必关心这个软件系统是如何实现的，采用了什么优秀的体系结构、先进的开发技术等，它只须关心这个软件能为最终用户提供什么功能，如何操作这个软件才能获得这个功能以及在使用软件的过程中应该注意的地方等。

对于建造完毕的房子，不会有人特意撰写一份使用说明书，因为房子对每个人来说都不陌生，其功能和使用方法不言而喻。由于房子是一个有形的实体，房子与房子之间的区别，只需要亲身去参观一次便可以一目了然。软件系统则不一样，随着应用领域的不同，各软件之间的差异也相当大，正如俗话说说的“隔行如隔山”。由于软件具有这样的特点，使得用户手册在软件产品中具有不可忽视的作用，用户手册成为推广软件的有力武器。

软件文档的格式也不是固定不变的，各公司可能有自己的一套标准，并且还可能和用户讨论形成新的软件文档结构。虽然如此，但是这些软件文档的基本内容是一致的。

1.4 软件文档必备的条件



软件系统配备软件文档不仅对于公司非常有益，而且也能够让客户从中受益。由于软件产品如何使用在某种程度上是要依赖软件文档来进行说明的，因此软件文档必须准确可靠。使用不准确的和已经过时的软件文档对于公司的发展也会产生一定的阻碍，同样也会对客户产生消极的影响。一旦客户发现在他们使用产品时遇到了问题，却不能通过求助于伴随软件产品的软件文档的手段进行解决时，客户就会对这种软件产品产生怀疑乃至失去信心，公司的信誉和利益自然而然地就会受到损害。

无论是面向客户的需求分析书、用户手册，还是面向开发者的概要设计书、详细设计

书、测试计划书，既然都是软件开发中产生的软件文档，则它们都具有一定的共性，而且都必须具有完整性、准确性、易用性和及时性。

1.4.1 完整性

大多数客户在刚开始使用一个新软件系统时，最快的入门方法就是软件的开发方派出技术人员对客户进行系统培训，通过从头到尾的演示过程，手把手地教会客户该软件的使用方法，并且及时回答客户的提问，这样的培训无疑是快捷而有效的。但是，对于大型的软件系统而言，其客户数量也是巨大的，不可能让所有的客户都停下手中的工作参与到软件的使用培训中去。通常只会选择让客户经理去参加，然后由客户经理再往下推广。这样的推广模式，首先不能保证直接参加软件使用培训的客户经理能充分地理解软件的使用方法；其次，客户经理在向下进行推广时，不能保证他们所说的都能完全符合该软件系统的实际情况。最后，最终用户怀疑自己理解的东西到底是不是开发方想传达的东西。只要传播的环节增加，讹传的可能性就会增加。如何才能判断自己理解的东西是不是正确的东西呢？依据就是软件文档。无论是直接得到软件开发方培训的客户经理，还是经由若干级传播之后的最终用户，他们看到的都是同一份软件文档。通过阅读相同的软件文档，不同用户之间的理解不会有太大的差异。而且将软件文档发送给最终用户让其自行阅读，比逐级进行培训可以节约大量时间和成本。

软件文档的作用是指导开发工作的进行，软件文档的内容应该是完整的。具体工作的展开都是基于软件文档的内容，软件文档记述了做什么、如何做等问题。软件文档就像小说一样，有头没尾、故事情节不完整的话，无法称其为一部完整的小说。每个阶段的软件文档必须为下一个阶段的工作提供必要的信息，这样才能让软件开发工作如流水般顺利地进行下去。

1.4.2 准确性

软件工程是一门工程学，它具备严谨、准确的特点。将这门学科运用到实际的软件开发工作中去，就要求软件开发的过程也必须是严谨而准确的。应用到软件文档的编写中就是，文字表述要明确，逻辑要清晰，不能出现含糊其辞的句子，不能提供多种方案供参考，文档中只能出现充分思考权衡之后的最终实现方案。这样的文档，才不会造成阅读者的歧义，不能像阅读文学作品那样，仁者见仁，智者见智，十个人读完之后有十种不同的感受。软件文档必须让每个人读起来得到的结论是一样的。

1.4.3 易用性

如果一套软件具备功能强大、界面美观、性能优良等优点，唯独操作过于复杂，这样的软件不容易得到推广。相反，在实现相同功能的前提下，操作简单的软件容易被客户接

受。软件文档也是一样的，必须具有易用性。软件文档的易用性是指易于查找，当客户想了解软件某方面内容时，必须很容易地知道在什么地方能获得自己想知道的内容，不能将时间花费在寻找上。

1.4.4 及时性

从决定开发一套软件系统开始，时刻都有变化的可能性，唯一不变的就是变更本身。前期终于和客户沟通完毕，形成了详尽的需求分析书，软件开发人员甚至已经开始着手设计该系统，即使是这个时候，也可能经常会收到来自客户的需求变更，于是需求分析书需要重新修改，按照需求变更的影响范围，概要设计书也必须相应地进行修正。经过一段时间的修改，概要设计书终于完成，进入了子系统和功能模块的详细设计阶段。但是，这时又可能发现之前的概要设计书某些方面不够合理，某些子系统的设计过于复杂，子系统间耦合度过高等问题，导致概要设计书的再次修改。随着每一次的软件文档修正，都会产生一个新的软件文档版本，我们必须保证的是，软件开发人员的工作必须是建立在最新软件文档的基础上，避免因为软件文档的版本问题导致返工。

所以，任何软件文档都必须及时地反映开发过程中的变更，并且及时地反馈到软件开发的相关人员手中。

1.5 软件文档的管理



1.5.1 管理者的作用

软件文档管理的标准是为对软件或基于软件的产品开发负有职责的管理者提供软件文档的管理指南，目的在于协助管理者在他们的机构中产生有效的文档。

管理者需要严格要求软件开发人员和编制组完成文档编制，并且在策略、标准、规程、资源分配和编制计划方面给予支持。

(a) 管理者对文档工作的责任

管理者要认识到正式或非正式文档都是重要的，还要认识到文档工作必须包括文档计划、编写、修改、形成、分发和维护等各个方面。

(b) 管理者对文档工作的支持

管理者应为编写文档的人员提供指导和实际鼓励，并使各种资源有效地用于文档开发。

(c) 管理者的主要职责

- (1) 建立编制、登记、发布系统文档和软件文档的各种策略。
- (2) 把文档计划作为整个开发工作的一个组成部分。
- (3) 建立确定文档质量、测试质量和评审质量的各种方法的规程。

- (4) 为文档的各个方面确定和准备各种标准和指南。
- (5) 积极支持文档工作以形成在开发工作中自觉编制文档的团队风气。
- (6) 不断检查已建立起来的过程，以保证符合策略和各种规程并遵守有关标准和指南。

通常，项目管理者在项目开发前应决定如下事项：

- 要求哪些类型的文档
- 提供多少种文档
- 文档包含的内容
- 达到何种级别的质量水平
- 何时产生何种文档
- 如何保存、维护文档以及如何进行通信

如果一个软件合同是有效的，应要求文档满足所接受的标准，并规定所提供的文档类型、每种文档的质量水平以及评审和通过的规程。

1.5.2 制订文档编制策略

文档策略是由上级(资深)管理者制订的，对下级开发单位或开发人员提供指导。策略规定主要的方向不是做什么或如何做的详细说明。

一般说来，文档编制策略陈述要明确，并通告到每个人且理解它，进而使策略被他们贯彻实施。

支持有效文档策略的基本条件有以下几点。

(a) 文档需要覆盖整个软件生存期

在项目早期几个阶段就要求有文档，而且在贯穿软件开发过程中必须是可用的和可维护的。在开发完成后，文档应满足软件的使用、维护、增强、转换或传输。

(b) 文档应是可管理的

指导和控制文档以获得维护，管理者和发行专家应准备文档产品、进度、可靠性、资源、质量保证和评审规程的详细计划大纲。

(c) 文档应适合于它的读者

读者可能是管理者、分析员、无计算机经验的专业人员、维护人员、文书人员等。根据任务的执行，他们要求不同的材料表示和不同的详细程度。针对不同的读者，发行专家应负责设计不同类型的文档。

(d) 文档效应应贯穿到软件的整个开发过程中

在软件开发的整个过程中，应充分体现文档的作用和限制，即文档应指导全部开发过程。

(e) 文档标准应被标识和使用

应尽可能地采纳现行的标准，若没有合适的现行标准，必要时研制适用的标准或指南。

(f) 应规定支持工具

工具有助于开发和维护软件产品，包括文档。因此尽可能地使用工具是经济的、可行的。

1.5.3 制订文档编制标准和指南

在一个机构内部，应采用一些标准和指南：

- 软件生存期模型
- 文档类型和相互关系
- 文档质量

这些标准和指南决定如何实现文档任务，将提供一些准则以评价机构内所产生的软件文档的完整性、可用性和适合性。

尽可能地采用现行的国家和国际标准，若现行的标准不适用，机构应制订自己的标准。

1. 选择软件生存期模型

现有的一些软件生存期模型，对于不同的阶段有不同的词汇，从软件文档的观点来看，采用哪种模型都无关紧要，只要阶段和相应的文档是清晰定义的、已计划的，并且对于任何具体软件项目是能遵循的。因此，管理者应选择一个软件生存期模型并保证该模型在他们的机构内是适用的。

管理者会发现所进行的阶段和相应任务的定义有助于监控软件项目的进展。特定阶段生成的文档可用做该阶段的评审、通过和完成的检验点，而这种检验应在下一阶段开始前进行。

2. 规定文档类型和内容

下面给出软件文档主要类型的大纲，这个大纲不是详尽的或最后的，但适合作为主要类型软件文档的检验表。而管理者应规定何时定义他们的标准文档类型。

软件文档归入如下三类别：

- 开发文档——描述开发过程本身
- 管理文档——记录项目管理的信息
- 用户文档——记录用户所需的信息

3. 确定文档的质量等级

仅仅依据规章、传统的做法或合同的要求去制作文档是不够的。管理者还必须确定文档的质量要求以及如何达到和保证质量要求。

质量要求的确定取决于可得到的资源、项目的大小和风险，可以对该产品的每个文档的格式及详细程度做出明确的规定。

每个文档的质量必须在文档计划期间就有明确的规定，文档的质量可以按文档的形式

和列出的要求划分为4级。

- 最低限度文档(1级文档): 1级文档适合开发工作量低于一个人月的开发者自用程序。该文档应包含程序清单、开发记录、测试数据和程序简介。
- 内部文档(2级文档): 2级文档可用于在精心研究后被认为似乎没有与其他用户共享资源的专用程序。除1级文档提供的信息外,2级文档还包括程序清单内足够的注释以帮助用户安装和使用程序。
- 工作文档(3级文档): 3级文档适合于由同一单位内若干人联合开发的程序,或可被其他单位使用的程序。
- 正式文档(4级文档): 4级文档适合那些要正式发行供普遍使用的软件产品。关键性程序或具有重复管理应用性质(如工资计算)的程序需要4级文档。4级文档遵守GB 8567的有关规定。

质量方面需要考虑的问题既要包含文档的结构,也要包含文档的内容。文档内容可以根据正确性、完整性和明确性来判断。而文档结构由各个组成部分的顺序和总体安排的简单性来测定。要达到这4个质量等级,需要的投入和资源逐级增加,质量保证机构必须处于适当的行政地位以保证达到期望的质量等级。

1.5.4 文档编制计划

文档计划可以是整个项目计划的一部分或是一个独立的文档。应该编写文档计划并把它分发给全体开发组成员,作为文档重要性的具体依据和管理部门文档工作责任的备忘录。

对于小的、非正式的项目,文档计划可能只有一页纸;对于较大的项目,文档计划可能是一个综合性的正式文档,这样的文档计划应遵循各项严格的标准及正规的评审和批准过程。

编制计划的工作应及早开始,对计划的评审应贯穿项目的全过程。如同任何别的计划一样,文档计划指出未来的各项活动,当需要修改时必须加以修改。导致对计划做适当修改的常规评审应作为该项目工作的一部分,所有与该计划有关的人员都应得到文档计划。

文档计划一般包括以下几方面内容:

- (a) 列出应编制文档的目录。
- (b) 提示编制文档应参考的标准。
- (c) 指定文档管理员。
- (d) 提供编制文档所需要的条件,落实文档编写人员、所需经费及编制工具等。
- (e) 明确保证文档质量的方法,为了确保文档内容的正确性、合理性,应采取一定的措施,如评审、鉴定等。

(f) 绘制进度表，以图表形式列出在软件生存期各阶段应产生的文档、编制人员、编制日期、完成日期、评审日期等。

此外，文档计划规定每个文档要达到的质量等级，以及为了达到期望的结果必须考虑哪些外部因素。

文档计划还确定该计划和文档的分发，并且明确叙述与文档工作的所有人员的职责。

▶ 本章小结

本章主要概括性地介绍了软件文档的意义、作用和分类等相关知识，以及软件文档必需具备的条件。还介绍了软件文档的管理。

由于篇幅所限，本章涉及的内容不是具体的理论和应用，而是在软件文档的背景知识和应用范围等方面给读者提供一个总体印象。在后面的章节中，将详细阐述软件文档的写作规范，并提供丰富的案例分析帮助读者理解相关知识的使用。

参考文献

- [1] 文斌，刘长青，田原编著. 软件工程与软件文档写作. 北京：北方交通大学出版社，2005
- [2] 肖刚编著. 实用软件文档写作. 北京：清华大学出版社，2005
- [3] 郭庚麒，余明艳，杨丽编著. 软件工程基础教程. 北京：科学出版社，2004
- [4] 莱芬韦尔等著，蒋慧译. 软件需求管理用例方法. 北京：中国电力出版社，2004
- [5] Roger S. Pressman 著. 梅宏译. 软件工程：实践者的研究方法. 北京：机械工业出版社，2002
- [6] 杨文龙，古天龙编著. 软件工程. 北京：电子工业出版社，2004
- [7] 陈宏刚，林斌，凌霄宁，熊明华，张亚勤编著. 软件开发的科学与艺术. 北京：电子工业出版社，2002
- [8] 董荣胜，古天龙. 计算机科学与技术方法论. 北京：人民邮电出版社，2002
- [9] *Guide the SE Body of Knowledge*. IEEE-Trial version (version 0.95), May 2001
- [10] Robert C. Martin. *Agile Software Development, Principles, Patterns, and Practices*. Pearson Education Inc., 2003