**北京邮电大学软件学院**

**2019-2020学年第一学期实验报告**

**课程名称：** 软件工程理论与实践

**项目名称： 实践一**

**项目完成人：**

**姓名：**  平雅霓 **学号：** 2017211949

**指导教师：** 杨金翠

**日 期： 2019年 11 月 7 日**

目录

[1. 问题 3](#_Toc27585)

[1.1. 如何知道承包方是否有能力完成这样的项目？ 3](#_Toc22554)

[1.2. 如何监控承包方的项目时间进度、工程质量、经费使用情况？ 3](#_Toc8695)

[1.3. 如何给承包方付款，降低风险？ 4](#_Toc19607)

[1.4. 怎么检查或评价承包方在工程工作中间的成果质量，并依据当前进度和工程成果，说明后续的风险？ 4](#_Toc21826)

[1.5. 进一步提出本项目的质量或可信赖性要求 7](#_Toc32269)

[1.6. 说明如何寻找承包商？如何监控承包商？ 8](#_Toc26106)

[1.7. 说明如何交付该项目，并在项目投入运行后，很好地维护系统的运行。 9](#_Toc18230)

[2. 模型与评价要求 9](#_Toc22700)

[2.1. 选择的模型 9](#_Toc17470)

[2.2. 选择此模型的原因 10](#_Toc9050)

[2.3. 检查点和被检查项 10](#_Toc23424)

[2.4. 交付的中间产品的基本要求 11](#_Toc16848)

[（1） 系统分析和设计阶段的产品质量评价 11](#_Toc29960)

[（2） 软件需求分析阶段的产品质量评价 12](#_Toc12689)

[（3） 概要分析阶段的质量评价 12](#_Toc8021)

[（4） 详细设计阶段的质量评价 12](#_Toc14881)

[（5） 编码和单元测试阶段的质量评价 13](#_Toc4841)

[（6） CSC集成和测试阶段的质量评价 13](#_Toc19645)

[（7） CSCI集成和测试阶段的质量评价 13](#_Toc21422)

[2.5. 给出对这些文档进行评价的标准，以便于专家组(或独立的软件工程检查小组)能客观地进行评价，防止人为因素的干扰(如腐败等)。 13](#_Toc24259)

# 问题

## 如何知道承包方是否有能力完成这样的项目？

在一般的软件采购的选择承包商阶段，在客户方进行初步需求分析时,可以邀请一些潜在承包商参加，候选的承包商将客户方发布的需求转换为详细的开发计划。此计划书中要描述系统的设计、规模、结构、复杂程度和其他信息，并需要提供项目费用和进度,给出技术需求条款、开发环境、开发工具、人员、管理以及其他组织结构和技术问题。在收到提案之前，甲方要确定评比标准，对承包商的组织能力、历史性能、费用估计方法以及其他性能估计的准则做出定义。此外，还要考虑承包商宣称的技术、开发过程和开发能力是否切合实际。

1. **判断承包商组织能力：**

承包商需要在软件项目开发过程中具有良好的组织能力，包括如何组织开发人员进行分工合作，如何将用户和开发方联系起来，如何开展需求分析、文档编写、代码编写、测试集成等工作，这需要承包商有良好的组织能力和历史经验，组织能力的判断可通过其本项目的计划书和历史项目记录来判断。

1. **判断承包商技术是否符合要求：**

对于本项目来说，由于计算精确性和武器装备的“任务关键”和“软件紧急修改”特征，我们需要承包商能高度重视项目的开发，尤其是在计算的精确度上，也就是说，技术层次是我们军方首要的关注点，因为用于军事的系统不容许有半分差错，否则将造成很大的人员伤亡和损失，因此这个要求需要承包商有很高精尖的技术来实现。选择承包商时，我们需要根据其以往的历史项目的性能来评判其计算水平是否达到了要求。

另外，只要求高计算性能是远远不够的，在战场上不可以关键时候掉链子，所以软件还需要有很高的可靠性。在系统装列后，若是战时状态，软件需要能无故障使用或故障率达到最低，具有良好的可用性和可靠性。一旦发生故障，承包商要尽快排除；若在和平时期，软件亦经常需要维护和存储装备系统，这对可维护性有了更高的要求。这也属于承包商的技术能力。

## 如何监控承包方的项目时间进度、工程质量、经费使用情况？

1. **监控项目时间进度**

监控时间进度可以设置检查点来进行监控，根据计划书中安排的大致时间流程来定期检查，比如何时完成项目大致的需求分析，何时进入编码阶段，一共需要几次迭代，每次迭代合何时结束等等，根据项目计划书的时间安排和实际情况相结合来进行定期检查。

1. **监控工程质量**

监控工程质量可能需要检查中间产品来实现，一个国防系统的成功研制应及时确保软件承包商按时交付质量可信的软件，因此要对软件开发过程中的提交物有进行检查和评审，而不是等到期时一次性验收。

1. **监控经费使用情况**

监控经费使用情况可以在进行中间阶段检查时施行，这样可以根据检查点的中间产品来大致估计出应当使用的价格，再与实际花费的价钱做比较，如此来判断此次检查前承包方花费的费用是否符合实际情况，是否其中存在贪污、薅羊毛的行为，这样也可以降低甲方的风险。

## 如何给承包方付款，降低风险？

在选择完承包商之后，可以先根据其计划书中所制定的价格交付一笔定金，比如交付估计总费用的20%作为定金，之后在每次的定期检查时，按阶段交付部分费用，如此按中间产品的检查来付费直到项目结束，最后再结清余款。如此可以降低承包商无法完成项目或者需要移交项目时产生的风险。

## 怎么检查或评价承包方在工程工作中间的成果质量，并依据当前进度和工程成果，说明后续的风险？

1. **如何检查、评价中间成果质量：**

可以在检查点检查中间产品。

评价中间产品的质量需要制定一个评价标准。对中间产品的评价相应的准则分为一般准则和每个阶段特定的准则。**一般准则为:①中间产品必须遵循所要求的格式；②与合同要求保持一致；③内部一致性；④可理解性；⑤技术的充分性；⑥各个阶段的完整程度。**

其具体的定义为：

1. **内部一致性**
2. 在文档中没有互相矛盾的两条语句。
3. 在文档中，一个给定术语、简称或缩略语始终指相同的事情。
4. 文档中对一个给定项目或概念始终采用相同的名字或叙述。
5. **可理解性**
6. 文档符合有关样本手册大写、标点符号和注解的规定。
7. 所有未包含在国防标准和国家标准中的术语均已定义。
8. 所有简称和缩略语在第一次使用时，其前面均使用全称，除非第一次是出现在表、图示和插图和公式中，但在正文或脚注中已给予解释。
9. 对所有的表、图示和插图在其出现之前，均已在正文中按它们出现的顺序引用过。
10. **对指定文档的可追踪性**

可追踪性有5个要素

1. 该文档包含或执行了原有文档中的合适的规定。
2. 这些文档中同一术语 简称或缩略语临相间的事情。
3. 这些文档中对同一项目或概念采用相同的名称或叙述。
4. 后继文档里的所有材料在原有文档里均有根据，没有引入不可追踪的材料。
5. 两个文档不互相矛盾。
6. **与指定文档的一致性**

文档之间的一致性指两个或多个非层次相关的文档不互相矛盾。

1. **适当的分析、设计或编码技术**

其要求为：在合同中应包括需求分析、设计和准备采用的编码技术的规定。承制方的软件开发计划(SDP)要闸述承制方实施这些技术的建议，包括在合同和软件开发计划中规定的技术是否符合合同的要求。

1. **空间和时间资源的适当分配**
2. 分配给指定元索的内存和时间的数量不超过文档中规定的该元素的限制。
3. 对某项下属元素的分配量的总和应在该项的总分配量之内。
4. **对要求条款的充分测试覆盖**
5. 对规定的每个需求至少进行一次测试。
6. 测试用例选用既有“平均”情形也有“极限”情形，如最小值和最大值。
7. 已选择“强化”用例，如超出范围的值。
8. 包含应用不同功能组合的测试用例。

**还可以包括质量因素的充分性、需求的可测试性、数据定义和数据使用的一致性、测试用例和测试过程的充分性、测试的完整性、回归测试的完整性等。**

其具体含义如下：

1. **质量因素的充分性**

该准则主要用于对“软件需求规格说明”中的质量因素要求进行评价。

1. **需求的可测试性**

其含义为:如果能设计一个客观的且可行的测试来确定软件是否满足需求，那么就认为这个需求是可测试的。在需求分析中所提出的任何要求条款都应当做到可测试。

1. **数据定义和数据使用的一致性**

该准则主要用于评价设计阶段的文档，要求每个数据元素的定义与其在软件的使用相一致。其目的是为了保护变量的量纲和值域范围，防止变量的越界和物理量纲转换时的错误。

1. **测试用例和测试过程的充分性**

测试用例和测试过程的设计要充分，要求每个测试用例和测试过程应确切地规定所提供的输入、跟随的步骤、预期的输出和评价输出结果所用的准则。如果这些因素中的任何一个没有规定,则测试用例或测试过程就是不充分的。

1. **测试的完整性**

如果所有的测试用例和所有的测试规程都已完成,所有的结果都已记录,所有的验收准则都已满足,那么测试就是完整的。

1. **回归测试的完整性**

在测试过程中，常常需要多次重复测试或重新测试。通常重复测试是为了改正前次测试发现的问题而进行的重新测试，重新执行测试用例和测试规程可能是一个子集,因此也称为回归测试。

1. **并依据当前进度和工程成果，说明后续的风险**

**可能存在的风险有：**

过度的行政管理性工作将太多的精力放在监督合同上，而忽视了技术方面的管理，最终导致技术上（比如计算的精度等必须达到的技术水平）未实现要求。

对项目进展和整体性能的测量是定性的，而缺乏定量的度量指示器，给出的分析太粗糙，无法准确判读。

缺乏最终用户参与，不能从最终用户的角度看待软件产品的功能性和可用性。

沟通不够，缺乏有效的交通渠道，信息不能及时的传达给合适的人。

## 进一步提出本项目的质量或可信赖性要求

1. **质量要求**

国际标准化组织将质量定义为:一个产品和服务满足所说明和隐含的需求能力的特征和特性的总和。国家标准《信息技术 软件工程术语》将软件质量定义为:“软件产品中能满足给定需要的性质和特性的总体；软件具有所期望的各种属性的组合程度；顾客或用户觉得软件满足其综合期望的程度；软件在使用中满足顾客预期要求的程度。”GB/T 11457- 2006的质量定义反映了几个层面的质量含义: (1)需求的满足程度;(2)所期望的各种属性的程度; (3)用户的满意度; (4)满足使用要求等。

**软件的质量要求可以通过如下的属性来评判：**

1. **功能性**

功能性是指当软件在指定条件下使用时，软件产品提供满足明确和隐含要求的功能的能力。功能性可以进一步分解为以下子特性：（1）适合性（2）准确性（3）互操作性（4）密安性。

1. **可靠性**

可靠性是在指定条件下使用时，软件产品维持规定的性能级别的能力。

软件可靠性可以进一步分解为3个子特性：（1）成熟性（2）容错性（3）易恢复性。

1. **易用性：**

易用性是从用户使用观点来看的，指在指定条件下使用时，软件产品被理解、学习、使用和吸引用户的能力。

易用性可以分解为4个子特性：（1）易理解性（2）易学性（3）易操作性（4）吸引性。

1. **效率**

效率是指在规定条件下软件产品所占用的资原数量以及所提供的功能和性能的能力。包括（1）时间特性（2）资源利用性。

1. **维护性**

维护性是指软件产品可被修改和维护的能力。

包括(1)易分析性(2)易修改性(3)稳定性(4)易测试性。

**F.可移植性**

软件产品从一种环境迁移到另外一种环境的能力。包括(1)易安装性(2)共存性(3)易普换性。

1. **可信赖要求**

可信赖性的定义：使用基于计算机的系统的用户和该系统开发、运行和维护人员必须能够信任所建立和使用的系统。

可信赖性包括可靠性、安全性、可维护性、密安性和生存性。

可靠性、安全性、可维护性、密安性和生存性表达了对基于计算的系统的可信任程度的要求，是评价系统是否可信任的基本属性。其中的密安性进一步分解为信息完整性、可用性和保密性要求。

## 说明如何寻找承包商？如何监控承包商？

1. **如何寻找承包商：**

在一般的软件采购的选择承包商阶段，在客户方进行初步需求分析时,可以邀请一些潜在承包商参加，候选的承包商将客户方发布的需求转换为详细的开发计划。此计划书中要描述系统的设计、规模、结构、复杂程度和其他信息，并需要提供项目费用和进度,给出技术需求条款、开发环境、开发工具、人员、管理以及其他组织结构和技术问题。在收到提案之前，甲方要确定评比标准，对承包商的组织能力、历史性能、费用估计方法以及其他性能估计的准则做出定义。此外，还要考虑承包商宣称的技术、开发过程和开发能力是否切合实际。

在问题（1）中已经阐述了如何判断承包商组织能力、判断承包商技术是否符合要求，下面对如何判断承包商费用估计是否合理进行阐述。

一般来讲,承包商提供的信息中有些是中肯的，但也有许多信息会言过其实，因为承包商的目的是得到更高的利润，而甲方则是希望花费更小的成本来达到目的，因此，甲方需要过滤出中肯、有效信息。如何判别费用是否合理，需要甲方也进行些许市场调研来判断。

1. **如何监控承包商：**

监督和质量审查得过细，方面客户方会增加许多工作量,另一方面客户方需要具有一批对软件质量非常清晰的监督人员,会造成客户方的成本提高。委托独立的第三方监理机构对承包商进行监理是一种方法，客户方必须提供适当的经费。

## 说明如何交付该项目，并在项目投入运行后，很好地维护系统的运行。

1. **如何交付该项目**

从软件开发相销售方的角度看，其开发出的软件需要转移给客户方，并在客户方指定的场地部署和运行，以支持最终用户的使用。从客户方的角度看，如果软件开发方能将软件完全移交，并能够在软件移交和交付后，保证客户方不需要开发方的技术支持和服务，就意味着软件系统能够完美地运行。可惜这种情况几乎是不现实的。软件系统的运行和使用中出现故障错误，或者是其功能和性能不能满足客户原有的或不断新添的需求使很正常的事情。

软件最终用户不需要了解系统的构造，只要系统能很好地运行或使用即可，由于软件系统本身潜在的故障,或者没有完全满足用户的需求,因此就需要提供及时的技术支持、保证在使用中遇到问题时能够得到及时的响应和问题的解决，以便系统能可靠地保证对最终用户的服务。

从软件运行和支持者角度看,为支持软件系统正常的运行，必须配备相应的人员、资源和环境,并对系统的运行规程和活动做出相应的要求。

提交一个用户完全满意的、终身不会出错的软件系统是不可能的。软件系统的维护和修改是必然的，因此必须在软件设计和编码阶段全面地考虑未来的修改和维护问题。提供足够的文档，并降低软件系统的维护成本。

1. **如何维护系统运行**

在传统工程中，维护是指修理和更换已经损坏或磨损的部件。然而软件不会损坏更不会磨损。IEEE把软件维护定义为：在提交后，为纠错、改进性能或其他属性，或者适应环境变更要求，对软件系统或部件进行修改的过程。软件维护实际上是提供客户满意的服务。

# 模型与评价要求

## 选择的模型

假设我代表军方，我认为应当选择增量式模型进行软件开发。

## 选择此模型的原因

增量式开发需要在开发过程中先实现那些需求明确的增量，随着系统开发的进展，再将一些不明确的需求逐渐清晰起来。在此项目中，由于是军方的系统，其总的需求大致是明确的，并且为期三年，所以采用增量式模型开发是合适的。其次，因项目是为期三年的大工程，其一部分需求是明确的，一部分需求是不明确的，需要在开发过程中逐渐完善，因此开发过程中可以将系统划分为若干个功能的子系统，并且把需求明确的部分作为其先期开发的增量来优先开发，可以降低项目风险。

在增量式模型的软件开发过程中，可以将每次迭代的开发出的成果先测试和发布，便于在测试和试用过程中对需求不清楚的部分进一步明确，并将需求明确的部分先行测试和试用，如此既验证了需求也增加了测试的次数。因为是军方的系统，所以还要保证其高可靠性，不能在关键时刻出现问题，这是需要无数次的测试来满足的，增量式模型正好迎合了这项需求，并且对于提高系统的稳定性和可使用性是有利的。

因此，我认为应该选择增量式模型进行军方软件的开发。

## 检查点和被检查项

一个国防系统的成功研制应及时确保软件承包商按时交付质量可信的软件，因此应当能够对软件开发过程中的提交物有进行检查和评审，而不是等到期时一次性验收。

检查点和被检查项的设立需要对比硬件的研制过程，在软件的开发过程中设立适当的里程碑和基线，对提交物进行评审。

国防系统的的开发包括硬件、软件和系统的集成，在软件开发过程中对提交物进行评审可以设立3个基线：功能基线、分配基线和产品基线。根据软件开发阶段的生命周期可知，开发过程中需要硬件和软件开发同时进行，从系统概念开始，软件开发依次进入系统的软件需求分析->软件需求分析->概要设计->详细设计->编程、单元测试、部件集成->CSCI测试阶段，硬件开发依次进入系统的硬件需求分析->硬件需求分析->概要设计->详细设计->制造->硬件测试阶段，最终软件和硬件开发过程同时进入系统集成和测试阶段。

**根据以上描述的软件开发阶段的生命周期，可以设立如下几个检查点：**

在系统软件/硬件需求分析中进行系统需求检查（SRR），以免需求存在较大偏差。

在系统软件/硬件需求分析之后进行系统设计检查（SDR）。

在软件需求分析之后进行软件规格说明检查（SSR）。

在硬件/软件的概要设计之后概要设计检查（PDR）。

在硬件/软件的详细设计之后关键设计检查（CDR）。

在编程、单元测试、部件集成测试之后进行测试准备检查（TRR）。

在硬件测试/CSCI测试之后进行功能配置检查（FCA）。

在功能配置检查后进行物理配置检查（PCA）。

在物理配置检查后进行正式合格性检查（FQR）。

**被检查项为：文档、代码。**

中间产品是指在软件开发的各个阶段产生的人工制品，例如软件需求分析，文档设计，文档和源代码，因此我们的被检查项也应该是这几项。

由于主要的中间产品是文档，所以文档作为很重要的中间产品需进行检查。

在进行编码和单元测试之前要在每个阶段进行阶段性的文档检查，初步运行概念文档、完成的运行概念文档、软件顶层设计文档、接口设计文档、数据库设计文档、软件详细设计文档、软件开发文档等待。

在编码和单元测试阶段要进行代码的检查，例如源代码、目标码、修改后的目标码等等。

## 交付的中间产品的基本要求

对中间产品的评价相应的准则分为一般准则和每个阶段特定的准则。一般准则为：①中间产品必须遵循所要求的格式；②与合同要求保持一致；③内部一致性；④可理解性；⑤技术的充分性；⑥各个阶段的完整程度。

还可以包括：①质量因素的充分性；②需求的可测试性；③数据定义和数据使用的一致性；④测试用例和测试过程的充分性；⑤测试的完整性；⑥回归测试的完整性。

除此之外，还需对在以下阶段中进行质量评估。

1. **系统分析和设计阶段的产品质量评价**

在系统分析阶段和设计阶段，所依据的主要输入是与甲方签订的合同，包括技术性要求和非技术性要求，需要对系统/子系统设计文档、软件开发计划、概要的软件需求规格说明、概要的接口需求说明进行内部一致性、可理解性、对指定文档的可追踪性、与指定文档的一致性、适当的分析设计或编码技术、空间和时间资源的适当分配、对要求条款的充分测试覆盖的评价。

建议的度量指标有4个：

1. 
2. 
3. 
4. 
5. **软件需求分析阶段的产品质量评价**

在软件需求分析阶段，所依据的主要输入是系统的设计规范。软件开发小组必须依据系统分配给软件的要求，对软件需求进行功能和非功能的要求分析。

例如：

软件需求规格说明、接口需求说明需要对内部一致性、可理解性、对指定文档的可追踪性、与指定文档的一致性、适当的分析设计或编码技术、空间和时间资源的适当分配、对要求条款的充分测试覆盖这几项进行评价。

1. **概要分析阶段的质量评价**

在概要分析阶段，所依据的主要输入是软件的需求文档。

在此阶段，需对软件设计文档概要针对内部一致性、可理解性、对指定文档的可追踪性、与指定文档的一致性、适当的分析设计或编码技术、空间和时间资源的适当分配进行评价，另外还需从CSCI到CSC要求分配的充分性进行附加评测。

还需要对接口文档设计概要进行内部一致性、可理解性、对指定文档的可追踪性、与指定文档的一致性的评价。

对软件测试计划进行内部一致性、可理解性、对指定文档的可追踪性、与指定文档的一致性和数据记录整理和分析方法充分性的评价。

对CSC测试要求进行内部一致性、可理解性的评价。

1. **详细设计阶段的质量评价**

详细设计是对概要设计的细化。在此阶段中需要对软件设计文档详细设计、接口设计文档、软件测试计划、CSU测试要求和测试用例、CSC测试用例、CSU和CSC的软件开发文档、软件测试说明测试用例进行评价。与上面评测的内容相似，此处不再赘述。

1. **编码和单元测试阶段的质量评价**

编码和单元测试阶段的重点是代码的质量是否符合工程所期望的编码标准，以及单元测试是否充分。需要对源代码、CSU测试规程、CSC测试结果、CSC测试规程、CSU和CSC开发文档内容进行以上多方面的评价。

1. **CSC集成和测试阶段的质量评价**

在该阶段，要对CSC集成结果、软件测试说明（STD）测试规程、CSC测试结果、CSC测试规程、已修改的源代码、已修改的开发文档进行以上多方面的评价。

1. **CSCI集成和测试阶段的质量评价**

在该阶段，要对软件测试报告、已修改的源代码进行以上多方面的评价。

## 给出对这些文档进行评价的标准，以便于专家组(或独立的软件工程检查小组)能客观地进行评价，防止人为因素的干扰(如腐败等)。

1. 撰写认真，文档应具有统一性(所有内容都指向同一个清晰的主题)、一致性(所有内容的组织清晰合理，符合逻辑)和支持性(有具体的细节或案例)，没有各种文字错误。
2. 正确的格式化使用标题、列表、留白等机制，让文档适宜阅读。
3. 不同的情态动词“应当”、“必须”、“最好要”等说明了用户对软件功能或非功能要求的强烈程度，文档的编写者必须慎用和理解这些词汇的要求。
4. 文档与实际的修改和记录要一致，不可以凭空杜撰或与实际工程差别很大。
5. 文档应对软件开发计划、系统设计、需求规格说明、接口需求说明、软件测试计划等内容进行准确详细的记录。
6. 文档内容不相互矛盾。
7. 对文档责任到个人，明确哪些人员具有文档的撰写和修改权。