

## 软件学院《Rational 统一过程》报告

班级	11
学号	55171123
姓名	刘世喆

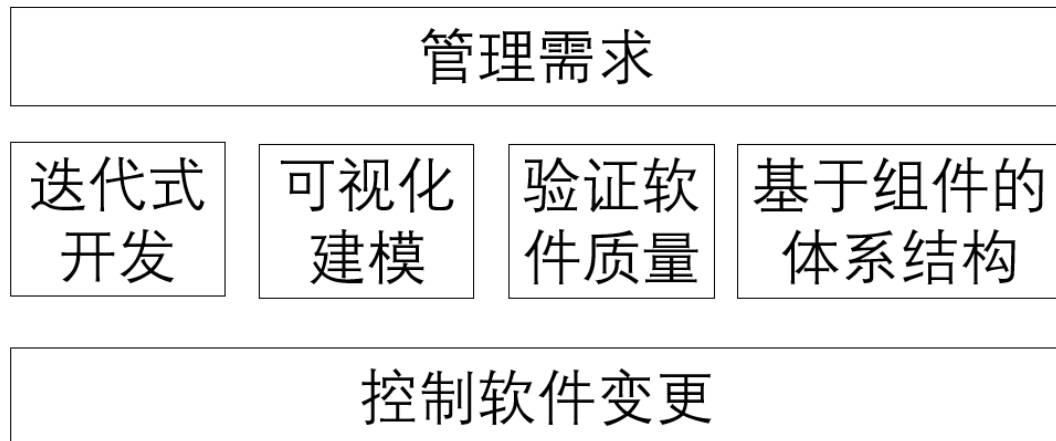
要求：报告用小四号字体，标清题号。不需要抄写题目。

## 一、 简答题(共计 55 分)

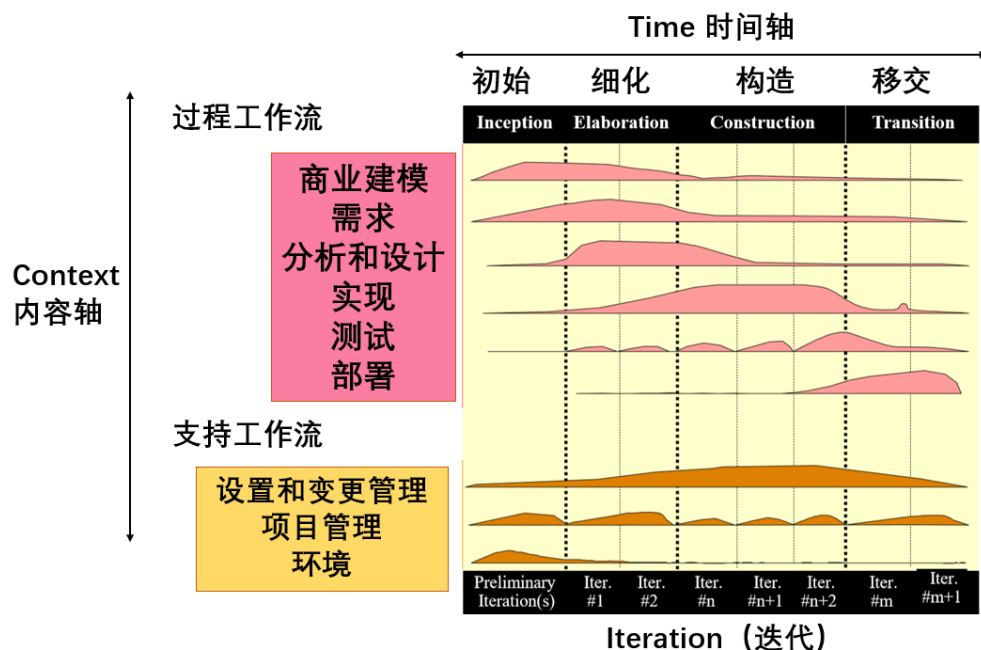
1.

六大最佳实践是：迭代式开发、管理需求、基于组件的体系结构、可视化建模、验证软件质量、控制软件变更。

它们之间的关系如下：



2. RUP 的二维过程结构图如下：



3. RUP 应用了四种重要的模型元素，分别是：

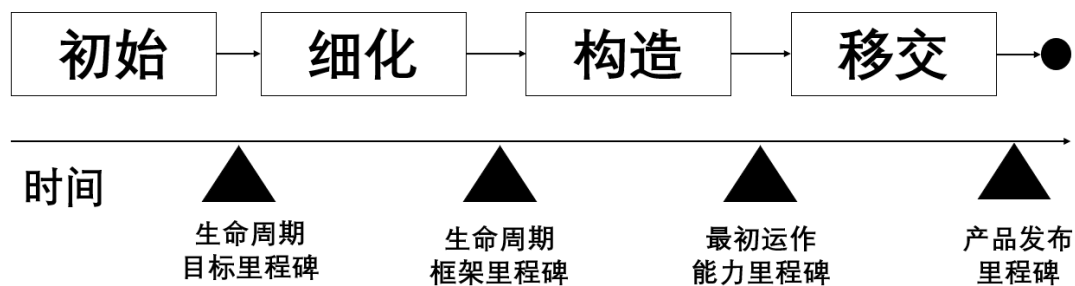
i. 工作人员(worker)：描述谁来做，具体来说工作人员(worker)定义了个人或一个工作组的行为和职责。

ii. 活动(activity): 描述怎么做, 活动(activity) 也即定义了 worker 执行的工作。

iii. 制品(artifact): 描述做什么。

iv. workflows(workflow): 描述什么时候做; workflows描述能够产出有用成果的有重要意义的活动序列, 并表示出 worker 之间的交互作用

4. (5 分) 请画出 RUP 的阶段和里程碑在时间轴上的分布顺序。



5. RUP 四个阶段的主要任务是:

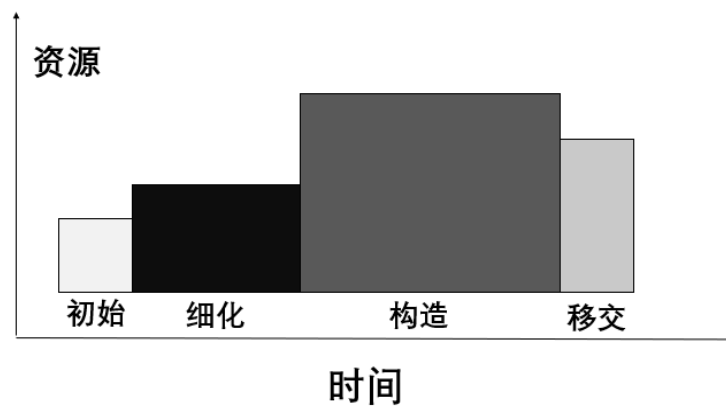
I. 初始(Inception)阶段: 确定最终产品的构想及其业务用例、并定义项目范围  
初始阶段以生命周期目标(LCO)里程碑为结束点。

II. 细化(Elaboration)阶段: 计划出必须完成的活动和需要的资源; 详细说明产品特性并设计架构  
细化阶段以生命周期构架(LCA)里程碑为结束点

III. 构造(Construction)阶段: 构造整个产品, 逐步完善视图、构架和计划, 直到产品(完整的构想)已完全准备好交付给用户。构造阶段以最初运作能力(IOC)里程碑为结束点。

IV. 移交(Transition)阶段: 移交产品给用户, 包括制造、交付、培训、支持及维护产品, 直至用户满意。移交阶段以产品发布版本里程碑为结束点, 这也是整个周期的结束点。

6. RUP 的典型项目剖面图如下：



项目各阶段持续时间和工作量的相对比例如下表：

阶 段	持续时间	工作量
初始阶段	10%	5%
细化阶段	30%	20%
构造阶段	50%	65%
移交阶段	10%	10%

7.

I. 框架就是当你去掉任何部分，就无法使其他人理解整个系统和解释它是如何工作的系统描述。对框架概念的解释：

i) 软件框架不仅仅注重软件本身的结构和行为，还注重其他特性：使用性、功能性、性能、弹性、重用、可理解性、经济和技术限制及折中方案、美学等。

框架是设计的一部分，它决定了如何建立系统。但它不是全部的设计，它只涉及一些重要元素——那些对系统质量有着普遍的深远影响的元素。

ii) 框架是关于结构和组织的，但是也处理行为。

iii) 框架不仅仅关心系统内部，还着眼于系统的两个外部语境：操作语境(它的最终用户)和开发语境（开发系统的技术人员）。它不仅包含系统的技术问题，还包含经济和社会问题。

iv) 框架同时还重视一些“软”问题，如风格和美学。

II. 模型和框架视图的区别：

i) 模型是系统的完整表达，不同的需求要创建不同的模型来表述

ii) 构架视图只关注对构架重要的方面

iii) 不是所有的设计都是构架

8.

I. 用例模型是需求工作流的结果。用例模型在不同阶段的作用是：

i) 在分析和设计阶段，用例是将需求活动 and 设计活动连接起来的桥梁，在用例实现中充当基础。通过遍历用例，开发人员会找到对象和类。

ii) 在实现阶段，设计模型是实现规格说明。用例是设计模型的基础，我们根据设计类实现用例。在设计模型中实现用例可以理解系统的动态性，并确定在哪里来优化系统性能。

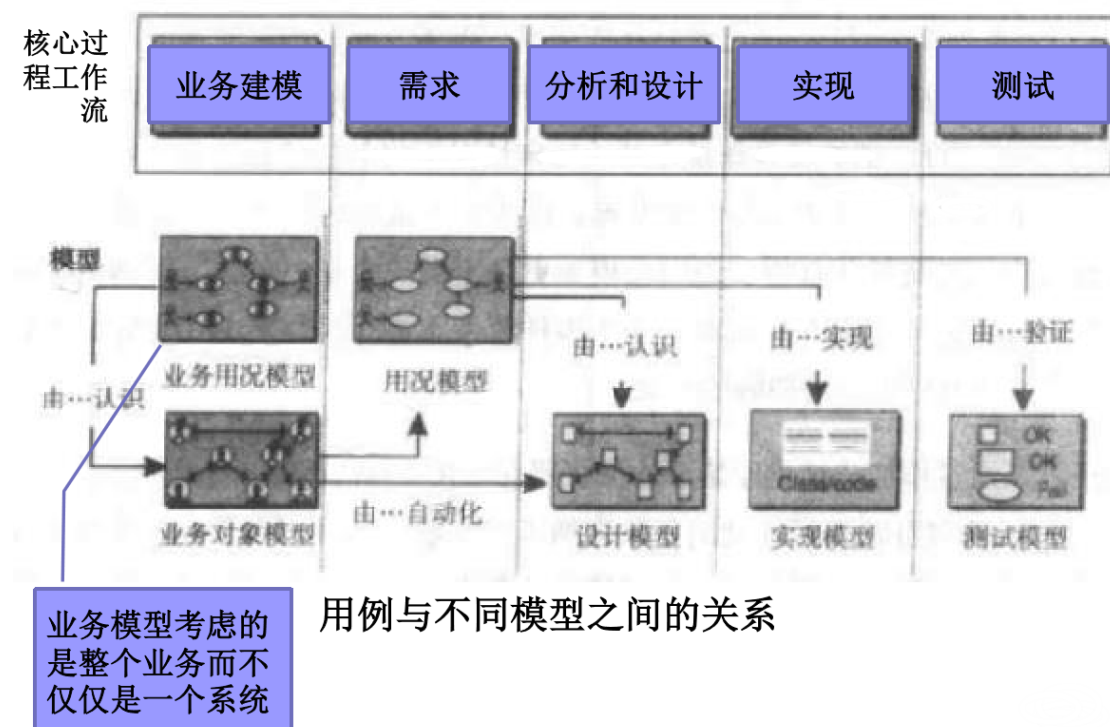
iii) 在测试阶段，用例构成了确定测试用例和测试规程的基础。每一个用例都用来验证系统。

iv) 在实施阶段，用例包可以用来计划阶段性实施，还可以定义系统变量

v) 用户界面的定义和原型设计也来源于以用例记事板形式出现的使用例

vi) 在业务建模中也使用用例概念。但是业务模型考虑的是整个业务而不仅仅是一个系统。

II. 用例模型与其他模型之间的关系图如下：



---

## 二、综合题(共计 45 分)

1) 分配项目进度表如下：

阶段	时间段	阶段持续天数	迭代个数
初始阶段	第 1 天-第 13 天	13 天	1 个
细化阶段	第 14 天-第 50 天	37 天	2 个
构造阶段	第 51 天-第 103 天	53 天	2 个
移交阶段	第 104 天-第 120 天	17 天	2 个

I.每个时间段安排解释：

一般项目各阶段进度安排持续时间比例大概是 10%、30%、50%、10%。由于公司之前已经存在一个电脑版软件，可以重用电脑版的所有服务接口，因此构造阶段持续时间比例需要减少；考虑到医疗方面软件要求准确度高，需要多分一些时间在测试上，部署时可能也要比一般软件费时，因此移交阶段分配 17 天比较好，此外多分配 1 天给细化阶段是为了多一些风险评估。

II.迭代个数解释：

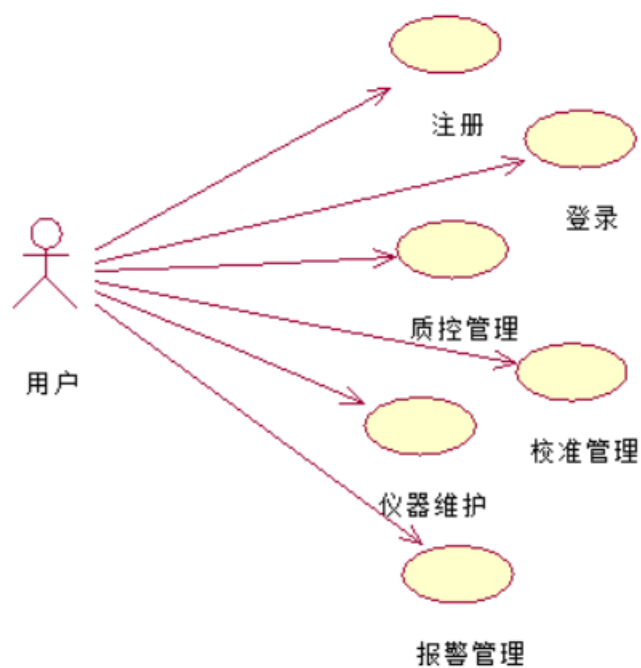
i)初始化阶段所需的一个迭代是为了建立原型，去验证设想的正确性、缓解主要技术风险；

ii)细化阶段所需 2 个是因为公司之前已经存在一个电脑版软件，有现成的构架；此外，该阶段没有太多的新因素要协调，开发的时候选择 Android Studio 工具即可；可能需要一些新的员工加入来解决医疗器材方面管理的问题。因此安排两个迭代是较好的。

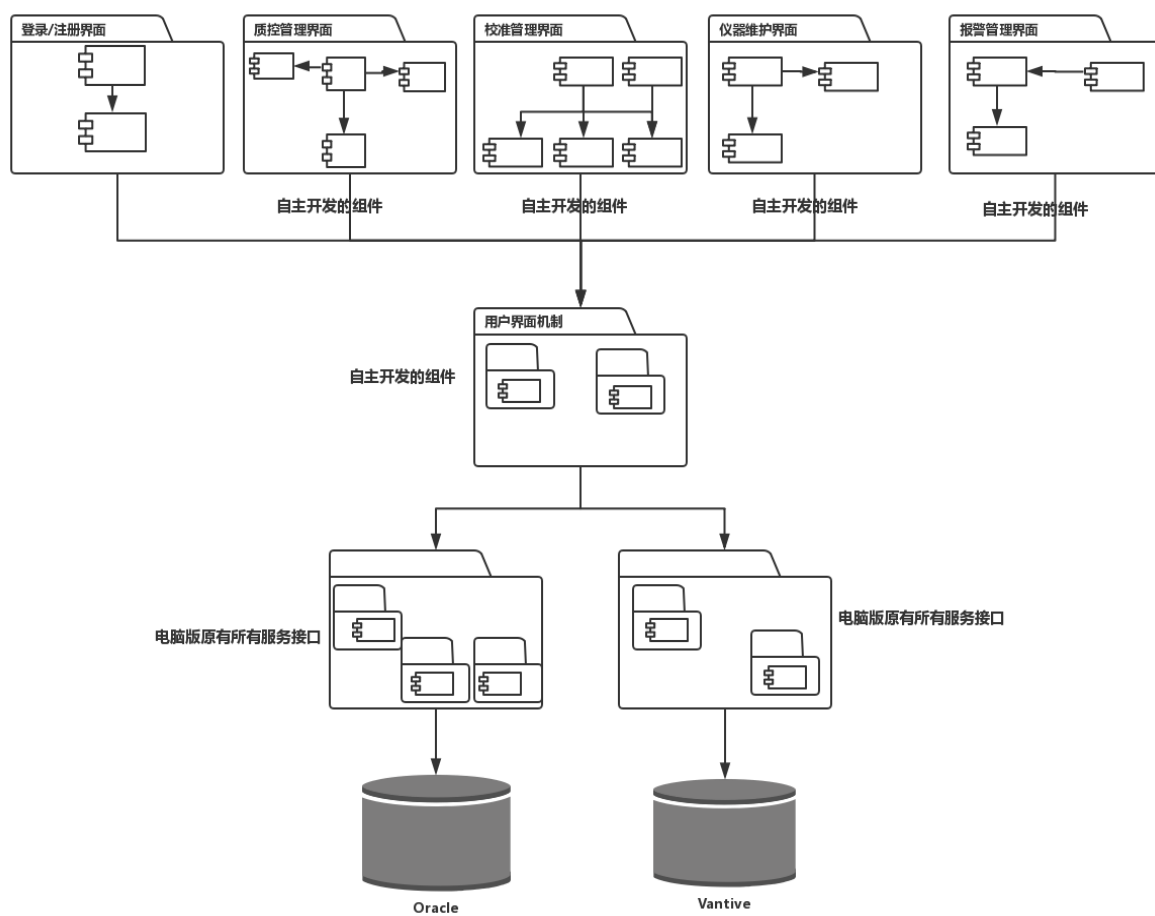
iii)构造阶段为了更好的集成和测试工作，2 个迭代比较合理。

iv)移交阶段 2 个迭代分别是：在 beta 版本之后发布最终版本迭代、由于市场的现实情况或最初版本的质量问题还需要迭代。

2) 系统的用例图如下：



3) 该软件的组件图如下：



---

4)

a) 答：

细化阶段会涉及的主要角色有五类，分别是：系统分析和设计师（包括了数据库设计师和封装设计师）、架构师、设计评审员、架构评审员、风险评估员。可能还会涉及到商业建模人员、项目经理、编码人员、测试人员等。

b) 答：

在这里，只考虑涉及到的主要角色，他们的工作任务如下：

i) 系统分析和设计师：寻找用例和参与者；定义职责、操作、属性、一个或多个类之间的关系，并决定如何调整它们使之适应实现环境，设计包或设计子系统，包括所有属于这些包或这些子系统的类；该过程的关键输出制品是设计模型。

ii) 架构师：设计架构，给出构架视图；该过程的关键输出制品是架构视图和软件构架文档。

iii) 设计评审员：评审系统分析和设计师的输出的关键制品；输入系统分析和设计师的设计模型，输出制品是分析设计模型的评审结果。

iv) 架构评审员：评审架构师给出的关键制品；输入制品是架构师的架构视图和架构文档，输出制品是架构方案评审结果。

v) 风险评估员：评估整个项目潜在的风险，输出制品是风险评估报告。

c) 答：

分析：每人的月成本 1.5 万，于是每人的天成本为 500 元。为了简化问题，在这里假设每类职员只安排了一人。

在（1）中，我们安排了 37 天，考虑设计和架构是这一阶段非常重要的一部分，因此安排的天数所占比重要大一些。

此外，架构师建立的是总体结构，关注的深度少，需要依赖一些系统分析和设计师的输出制品，他注重的角度是宏观的，因此所占用的天数会比系统分析和设计师少一些。

为了保证项目不会造成大的损失，安排一下风险评估也是非常有必要的。



---

根据上述分析，需要每个角色需要的工期和人员成本消耗如下表所示：

角色	天数（天）	人员成本消耗（元）
系统分析和设计师	20	10000
架构师	12	6000
设计评审员	4	2000
架构评审员	3	1500
风险评估员	5	2500

关于总天数的解释：总天数之和不是 37 天的原因是考虑了角色的并行过程，如系统分析和设计师、架构师两个角色工作日期是存在一定的重叠的。

根据上表，可以算总成本消耗（用  $\text{sumCost}$  表示），则

$$\text{sumCost} = 10000 + 6000 + 2000 + 1500 + 2500 = 22000 (\text{元})$$

对于迭代计划图，一般细化阶段的消耗比例为 30%，也就是说，这阶段细化阶段消耗的钱（用  $\text{planCost}$  表示）是：

$$\text{planCost} = 10 * 30\% = 3 (\text{万})$$

对于结果的解释：上述分析时，只考虑了主要角色，因此计算出来的  $\text{sumCost}$  值应该是要小于  $\text{planCost}$  的值的，也就是说，上表的安排是符合迭代计划图细化阶段的消耗比例的。