

华中师范大学计算机学院

实 验 报 告 书

实验题目： 华中师范大学计算机实验室预约管理系统项目集成计划

课程名称： 软件项目管理

主讲教师： 李蓉

文档编写人员：蒲东齐 周宏岳

实验时间： 2018 年 01 月 03 日

计算机实验室预约管理系统项目集成计划

1.	导言
1.1	目的
1.2	范围
1.3	缩写说明
1.4	术语定义
1.5	引用标准
1.6	版本更新记录
2.	项目概述
3.	项目任务范围和项目目标
3.1	项目目标
3.2	项目任务范围
4.	项目组织结构
5.	项目生存期
6.	时间计划
7.	项目规模估算
8.	质量管理计划
8.1	导言
8.2	组织机构
8.3.	质量目标
8.4.	质量策略
8.5.	质量保证活动
8.5.1.	项目审计
8.5.2	过程评审
8.6.	质量控制活动

8.7.	记录的收集、维护和保存
9.	项目成本估算
9.1.	组织及职责
9.2.	配置管理环境
9.3.1	配置库目录结构
9.3.2	用户及权限
9.4.	配置管理活动
10.	风险计划
11	质量度量
12	项目沟通与审核

1. 引言

1.1 目的

为了提高高效实验室的利用率，以及提高学生进行实验的积极性，同时提高实验室利用率，使同学们更合理的利用教学资源，我们决定开发实验室预约管理系统。该系统主要针对传统的人工管理效率低下、教学资源得不到有效利用等缺点。该系统主要实现了学生根据学习的实际情况进行教学实验的申请，查看当前实验室占用情况以及可预约时间。系统管理员管理用户信息、管理实验室开放时间以及发布系统公告，同时该系统提供了简单方便地用户界面。本系统主要使用 JavaScript、CSS、HTML 技术，实现前端功能。

1.2 范围

本文档定义了项目实施的方式和计划，未定义项目实施的过程规范和产品标准，有关内容可查阅企业的标准规范库。

1.3 缩写说明

CM: Content Management 的缩写。

KFL: Key Feature List 的缩写。

KPA: Key Practice Are 的缩写。

QA: Quality Assurance 的缩写。

SCM: Software Configuration Management（软件配置管理）的缩写，SW-CMM2 级 KPA 之一

SPT0: Software Project Tracking and Oversighting（软件项目跟踪与管理）的缩写，SW-CMM2 级 KPA 之一。

1.4 术语定义：

无。

1.5 引用标准

无。

1.6 版本更新记录

编写日期	版本	作者	说明
2017-12-14	0.1	蒲东齐	初稿
2017-12-16	0.2	蒲东齐	模块接口
2017-12-20	0.21	周宏岳	编审
2017-12-25	1.0	蒲东齐	定稿

2. 概述项目



本项目将开发出一个有效的实验室预约管理系统，使学校对实验室的管理更加科学更加规范化，同时能根据系统所提供的信息使实验室的安排更加的高效。传统的实验室管理方式，积累量大，易出错，在实验室开放方面，学生很难查看实验室的空闲时段，管理员也很难为学生预备实验条件。在选择实验课方面，总是以班级或者专业为单位进行排课，使得教学缺乏自主化，人性化，而且往往会师生之间相互不了解而产生的盲目教学等情况，影响学生、老师的积极性，从而影响实验教学的效果。

为了提高实验室的利用率，使实验室的安排更加合理化和有效性，建立一个信息化的实验室预约管理系统势在必行。因此本系统便应运而生，使用信息化和计算机管理技术，使得系统管理员能够管理教师和学生的相关信息，发布系统公告以及对实验进行管理；实验管理员能够在网上发布相关的实验，以及对教师和学生申请预约的实验进行在线的审核；教师和学生可以很方便地在网上进行实验室预约，并且查询实验的状态，及时了解实验内容。

系统采用 B/S 架构，符合操作简单，界面友好，灵活，实用，安全的要求，能够完成开放实验室预约管理的全过程。系统的最大特点就是打破了传统意义上的实验室管理方式，使得学生、老师在整个教学过程中更加自主，不断提高了两者的积极性，同时也提高了教与学的质量，提高了实验教学管理的效率。

3. 项目任务范围和实施目标

《华中师范大学计算机实验室预约管理系统》项目需要完成的任务总体分包括八大方面：个人信息模块，实验室预约模块，取消预约模块，联系管理员模块，可借教室信息查询模块，已借教室信息查询模块，实验室消息公告模块，超时记录信息查询模块。每一部分都是本系统的重要部分，能够最大限度的发挥作用，为学校的实验室管理提供支持。

按功能分为：

- 个人信息模块的设计与开发；
- 实验室预约模块的设计与开发；
- 取消预约模块的设计与开发；
- 联系管理员模块的设计与开发；
- 可借教室信息查询模块的设计与开发；
- 已借教室信息查询模块的设计与开发；
- 实验室消息公告模块的设计与开发；
- 超时记录信息查询模块的设计与开发；

按工作性质分为：

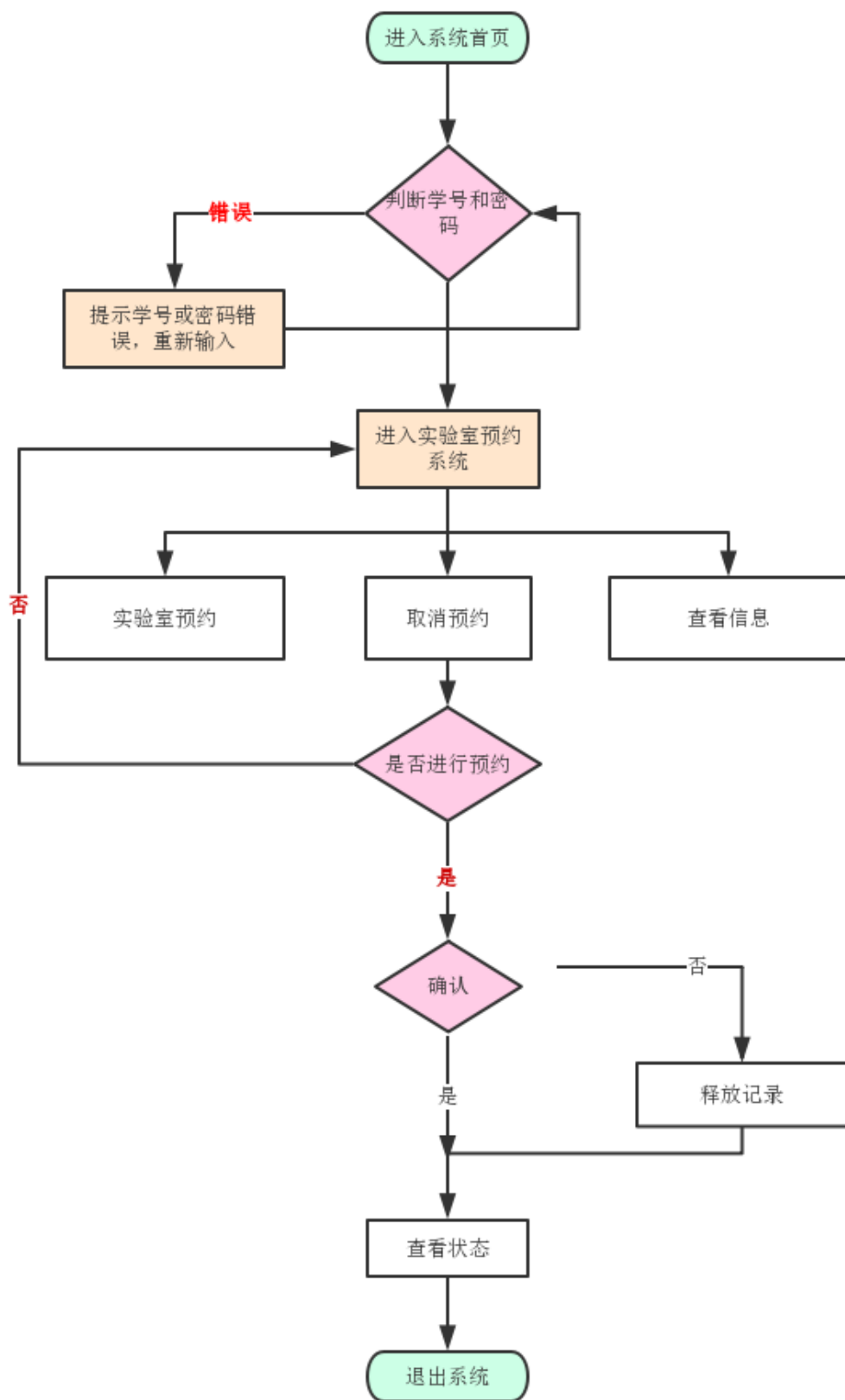
- 网页的设计与开发；
- 后台应用服务的设计与开发；
- 数据库的设计与开发；
- 系统内容的设计和维护；

提交给用户的产品包括：

- 平台应用系统软件包；
- 平台发布内容数据库；
- 产品录入维护工具包；
- 平台内容管理过程文档；
- 平台系统使用维护说明文档；

该项目实施的目标：

- 6 个月内完成平台的开发；
- 使教室，学生熟悉平台使用规则；
- 初步建成支持平台内容管理的工具集；



3.1 项目目标

建立一个合理的实验室预约管理系统，从而能够对实验室使用做完善的管理，使实验室管理更加科学规范，并能根据系统提供的准确信息进行适当的调整，使学校的管理更有条理化。

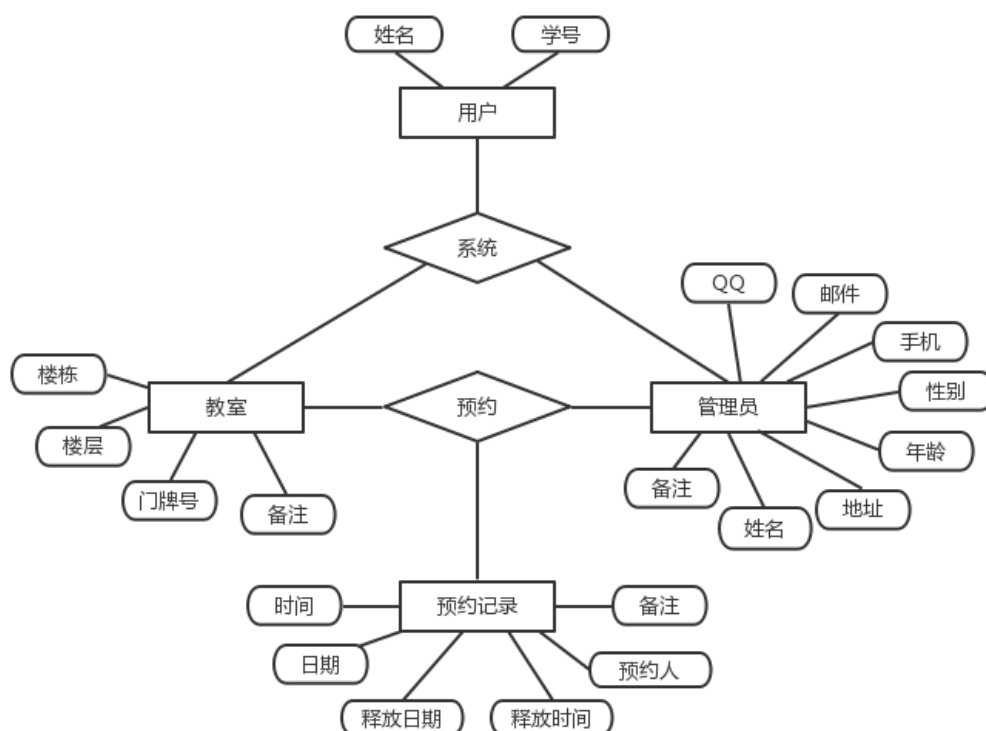
实验室预约管理系统软件与传统的人工操作来比，首先是信息的安全性有了明显的提高，其次是对信息的操作方面快捷，再次对信息的备份和恢复比较简单，而且节省了学校和管理方面的成本。

采用现有的软硬件软件及科学的管理系统开发方案，建立实验室预约管理系统，实现移动信息管理的计算机自动化。系统应符合学校实验室管理制度，并达到操作直观、方便、实用、安全等要求。并做到以下几点：

本项目的产品可以达到以下目标：

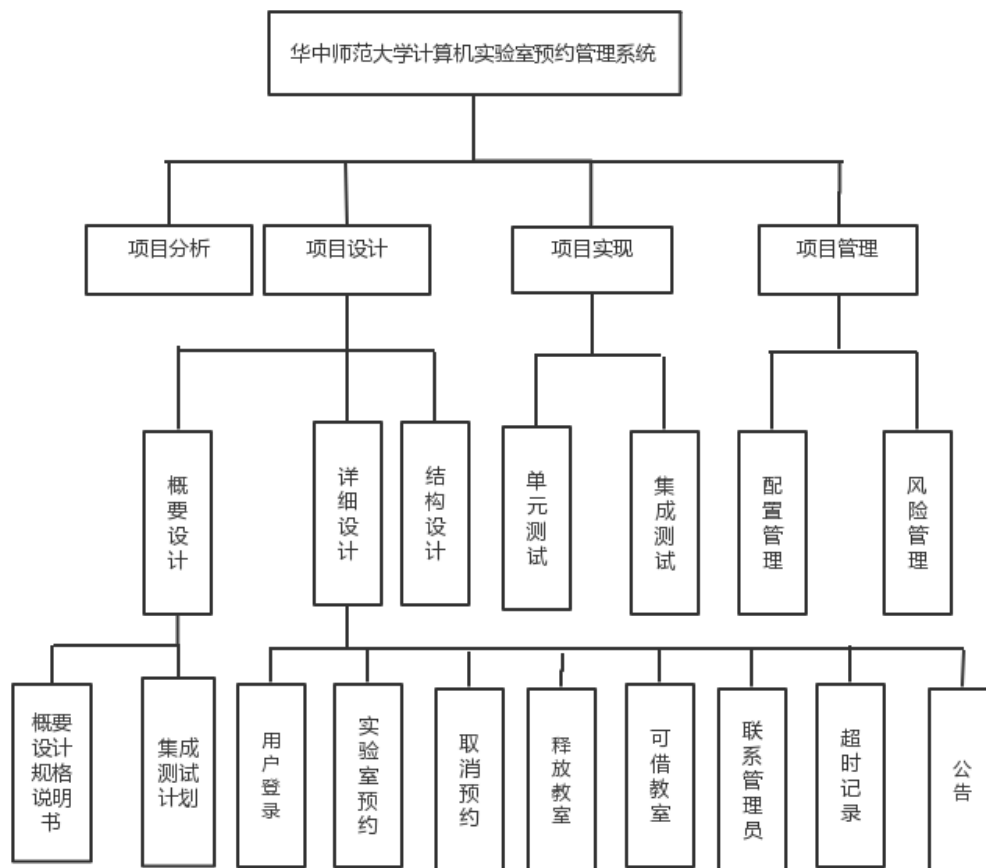
- 简单性（系统设计尽量简单，从而实现使用方便、提高效率、节省开支、提高系统的运行质量。）
- 灵活性（系统对外界的变化有较强的适应能力）
- 完整性（系统是各个子系统的集合，作为一个有机的整体存在。因此，要求各个子系统的功能尽量规范，数据采集统一，语言描述一致。）
- 可靠性高（实现安全的、可靠的数据保护措施）
- 存储量大保密性好
- 寿命长
- 成本低等。

实验室预约管理系统 8 大功能模块的目标：



- (1) “个人信息模块”：用于用户查看个人信息，如姓名，学号，学院，专业，预约教室信息，超时罚款信息等。
- (2) “实验室预约模块”：用于用户查询需要预约的楼栋，楼层，教室，日期，时间等。
- (3) “取消预约模块”：用于用户对已经预约好的教室进行取消操作。
- (4) “联系管理员模块”：用于用户对实验室管理员联系方式进行查看。
- (5) “可借教室信息查询”：用于用户对于当前可借实验教室进行查询。
- (6) “已借教室信息查询”：用于用户对于当前以借出教室进行查询。
- (7) “实验室消息公告模块”：用于用户对实验室最新的公告进行查看操作。
- (8) “超时记录信息查询”：对当前异常的实验室，如超时使用的实验室进行查询操作。

另外，通过项目进一步验证与完善该实验室预约管理系统，可以锻炼团队成员的编程能力与成员之间的协同精神。



3.2 项目实施策略

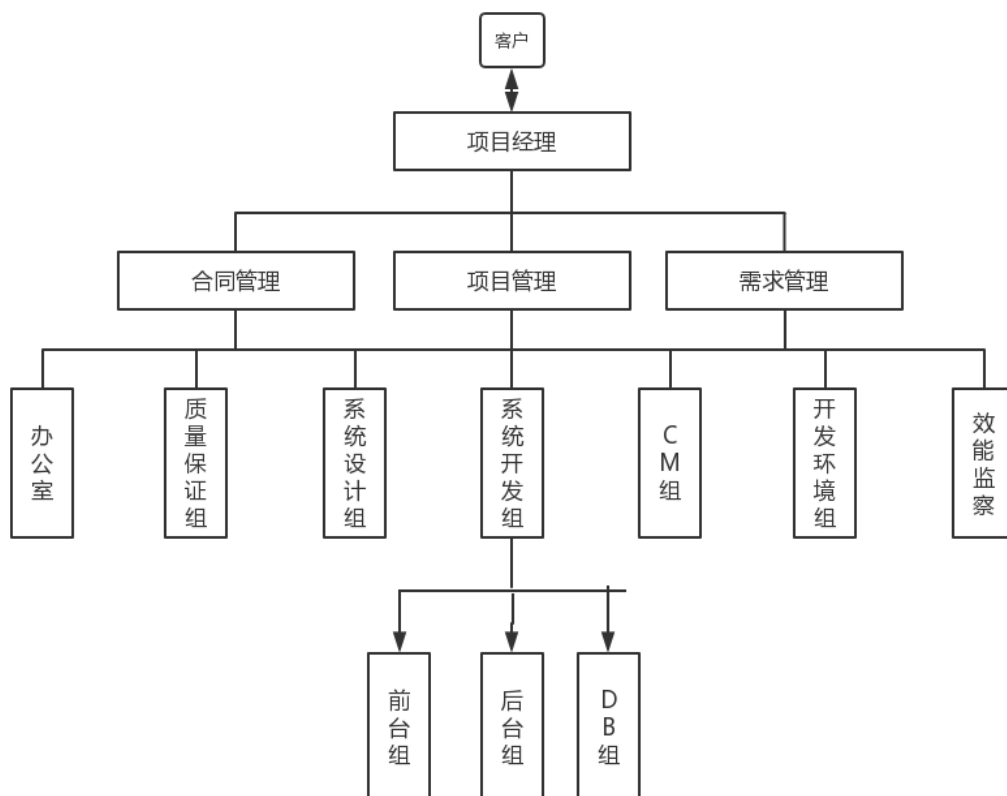
- 项目管理策略
 1. . 项目管理过程中应该遵循团队建立初期制定的项目管理过程规范。
 2. 根据项目原始计划中的评审点进行跟踪与管理，在与团队成员沟通后，项目主要管理人员可以根据结果对项目计划进行适当的调整与更改。
 3. 评审采用定期评审，阶段评审和事件评审相结合的方式。
 4. 每周进行一次团队小结，看各自完成的项目进度怎么样了，互相配合对法的进度。每月进行一次总体汇报，详细报告各自的工作进度与完成的指标。
- 软件开发策略
 - 1). 采用增量式模型逐步构建系统。
 - 2). 以周为阶段来提交任务。
 - 3). 开发软件时，团队成员的任务分配按照自己擅长的技术来分配。能者多劳，多劳者多得。
 - 4). 开发阶段，可以采用已有现存的模块，不必完全自己开发。

- 5) . 各个模块之间的接口应在最开始分配任务时就设定好，以免更改浪费时间。
- 6) . 在开发完成后，应该再对该系统进行多次测试，以免有漏洞。

- 质量保证策略

- 1) . 加强对团队成员的技术能力以及对保证质量的概念。
- 2) . 项目管理人员应加强对过程的控制，重点确定项目中应该控制的过程。
- 3) . 加强对产品规范的审计，重点确定该项目中需要审计的部分。
- 4) . 在开发完成后，应该再对该系统进行多次测试，以免有漏洞。
- 5) . 实施完整的软件配置管理。

4. 项目组织结构

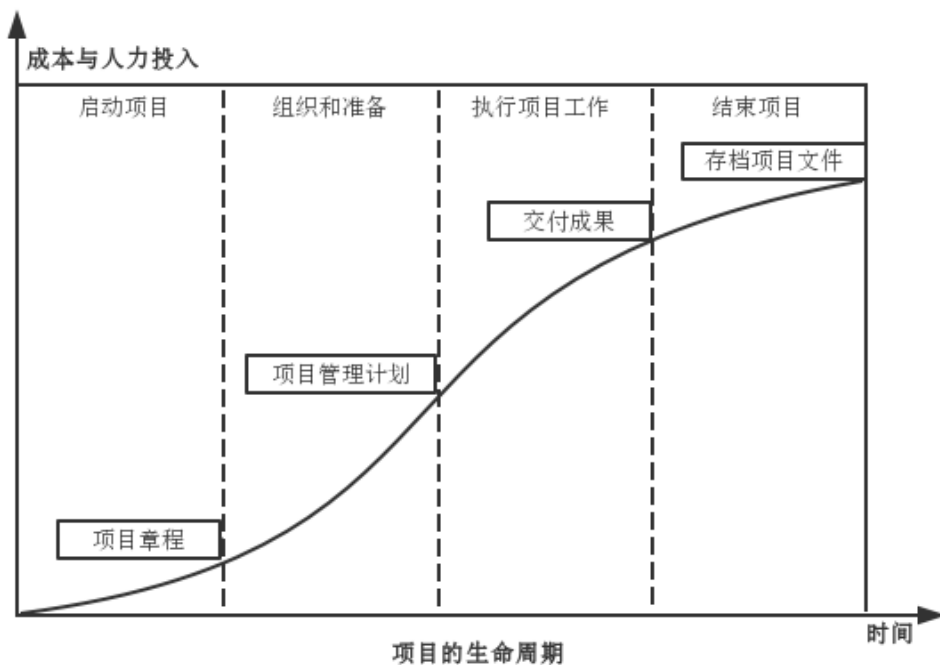


项目组织结构框图

由于该项目在实施过程中需要设计不同的各方面人才，而各人员之间的技术能力，任务以及职责，空闲时间都不同。因此明确定义项目组织结构和各自职责可保证项目的顺利进行。该项目的组织结构如下所示：

- 系统界面设计：
 - 负责编写出用户选择的界面
 - 负责把界面按钮连接到后台编写的程序
- 管理员模块的编写
 - 编写教师模块操作的源代码
- 学生模块的编写
 - 编写学生模块操作的源代码

5. 项目生存期



根据该项目的特点并结合已有的资料，本项目生存期采用生存期模型采用增量式模型。

生存期中的各个阶段定义如下：

- 学生模块设计：

阶段目标：编写学生模块操作的源代码

输入：系统设计说明书，数据库结构定义

过程：详细设计，编码，代码检查，代码评审，单元测试。

输出：详细设计说明书，源代码 1，可运行版本 1.

- 学生数据操作模块的编写

阶段目标：编写学生数据操作模块的源代码

输入：系统设计说明书，数据库结构定义

过程：详细设计，编码，代码检查，代码评审，单元测试。

输出：详细设计说明书，源代码 2，可运行版本 2.

- 管理员数据操作模块的编写

阶段目标：编写教师模块操作的源代码

输入：系统设计说明书，数据库结构定义

过程：详细设计，编码，代码检查，代码评审，单元测试。

输出：详细设计说明书，源代码 3，可运行版本 3.

- 实验室和教室数据模块操作

阶段目标：编写课程和专业数据模块操作

输入：系统设计说明书，数据库结构定义

过程：详细设计，编码，代码检查，代码评审，单元测试。

输出：详细设计说明书，源代码 1，可运行版本 1.

- 帮助文档编写

阶段目标：编写出帮助文档。

输入：系统设计说明书，数据库结构定义

过程：详细设计，跟踪源码，检查系统，测试。

输出：详细设计说明书，帮助文档。

- 产品提交

阶段目标：产品可投入使用

输入：系统软件包

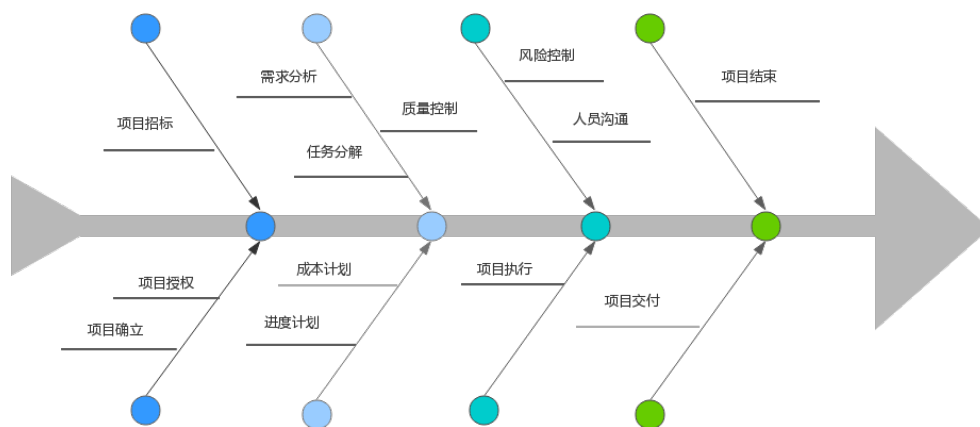
过程：产品提交

输出：验收报告

6. 时间计划

1. 初期项目计划

由于项目初期信息不是很充分，所以初期的项目计划只存在一个计划表格，相当于一个大计划，简单说明计划的执行步骤，如下表所示。

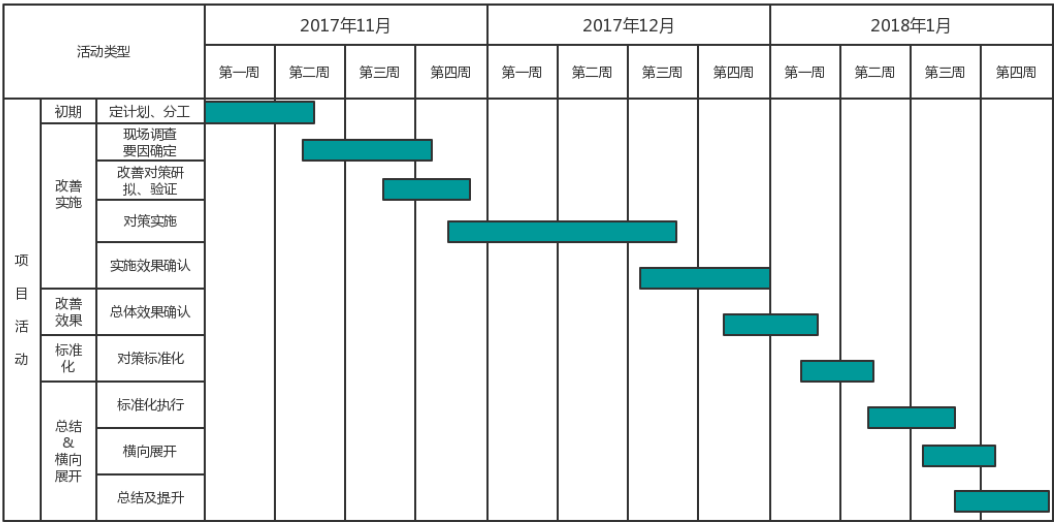


项目初期计划

2. 项目计划的细化

首先，根据 WBS 的分解情况，继续分解相应的活动（任务），使用 MS Project1 具，将分解后的所有活动（任务）和 WBS 的代码录入项目计划文件中，然后确定各个活动之间的关系。由于采用增量式的生存期模型，在需求设计之后，系统的功能采用增量方式实现，实施阶段分 4 个增量，对各个任务（活动）分配相应的资源，然后经过不断地资源调整优化以及工期、活动关系的调整等。再经过多次的评审讨论、最后计划通过评审，将此计划存为基准计划。项目的基准计划如下。

项目甘特图



3. 里程碑

项目负责人更加关注里程碑的发展，所以需要有一个里程碑的图示，如下表所示：

里程碑图示

事件	时间
学生模块-增量一	2017. 11. 15
学生数据操作-增量二	2017. 11. 30
管理员数据操作-增量三	2017. 12. 1
实验室使用情况操作-增量四	2017. 12. 15
系统 web 界面	2018. 1. 30

7. 项目成本估算

本项目采用自下而上和参数法综合的估算方法，具体过程如下：

（一）、分配任务前

开始分配任务的时候，根据不同人以往类似项目的经验，采用类比估算方法，进行粗略的估算，有的同学 JavaScript+CSS+HTML 的技术接触的比较早，相关项目经验也可以借鉴。根据 SOW 的说明，基本上需要 2 个开发人员，1 个月左右的开发时间，基本上是 2 人月的规模。

（二）、任务分配后

任务分配之后，根据现有的资源和 WBS 分解的结果，进一步细化估算，由于 WBS 分解是针对项目的功能进行的分解，在成本估算的时候，首先估算每个任务的开发规模，然后在通过系数获得相应的质量、管理任务的规模，从而计算直接成本，然后计算间接成本，以及总成本，具体过程如下。

（一） 人力资源：

2 个开发人员：

1 个项目管理人员

1 个配置管理人员

（二） 设备资源（作为间接成本计算）

2 台电脑

1 台服务器

项目规模估算表 注：规模单位为人/天

WBS	名 称	估 计 值(人天)	小 计(人天)	总 计(人天)
1.1	学生登录	3	3	3
1.2	查询实验室信息	2	2	5
1.3	查询可用时间	2	2	7
2.1	修改实验室信息	4	4	11
2.2	修改可用时间	1	1	12
3.1	管理员的登录	10	10	22
3.2	Web 界面构建	15	15	37
3.3	数据库连接及其他	23	23	60

估算步骤如下：

1. 获取项目分解结果 WBS

a) 任务分解是根据项目的功能进行分解的，

2. 计算开发成本

a) 由于任务分解的结果主要是针对开发任务的分解，管理任务和质量任务可以通过计算开发任务得到，根据以往经验，管理任务和质量任务=20%*开发任务。

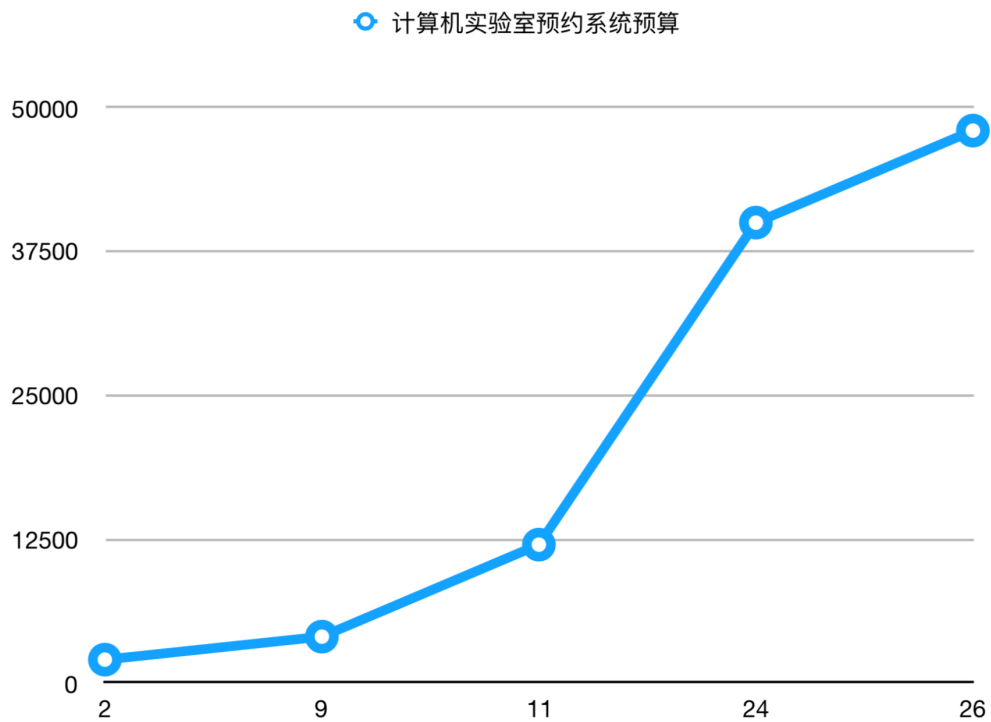
- b) 从表得知项目规模是 120 人天，开发人员成本参数 400 元/天，则内部的开发成本=400 元/天*120 天=48000 元，
3. 计算管理、质量成本
- a) 项目的管理和质量成本=开发成本*20%=9600 元，
4. 直接成本=48000+9600=57600 元，
5. 计算间接成本
- a) 间接成本包括前期合同费用、房租水电、培训、员工福利、客户服务等，
- b) 根据以往经验，采用公式：间接成本=25%直接成本=12000
6. 计算总估算成本
- a) 项目总估算成本=57600+12000=69600 元。
7. 重新评估项目的报价
- a) 重新评估一下项目的报价准确性，当然这时候，项目的合同已经签署了，报价是不能更改的，但是通过再次的评估可以进一步明确企业的项目运作和利润情况等，
- b) 如果项目的利润是 30%，其中风险基金 10%，利润 15%，税费 5%。则项目的总报价=69600*1.3=90480 元,,应该说报价还是比较合适的。

（三）成本预算

资源费用比例

资源名称	类型	缩写	最大单位	标准费率	加班费率	每次使用成本	成本累算	基准日历
蒲东齐	工时	蒲	100%	0.0 0元 /工 时	0.00 元/ 工时	0.0 0元	按比例	标准

周宏岳	工时	周	100%	0.0 0元 /工 时	0.00 元/ 工时	0.0 0元	按比例	标准
-----	----	---	------	----------------------	------------------	-----------	-----	----



8. 质量管理计划

8.1 导言

为保证项目最后的质量是可靠可用的，本项目制定了相关质量管理计划，这样既保证了相关责任落实到位，也使整个项目进程管理有据可依，同时也有利于项目有序按计划进行。

8.2 组织机构

在项目实施期间成立项目质量保证组织，该组织由质量保证人员和组长组成，组长负责质量监督工作及项目进展过程中各环节的质量把关，开发人员负责质量控制的工作，质量保证人员负责质量保证的工作。

组长的质量保证职责如下：

- ✦ 对项目实施情况进行监督，包括对项目实施过程和工作产品进行监督检查实施项目组成员的质量保证培训；
- ✦ 制定质量保证计划；

- ✦按计划实施审计活动, 依照质量保证计划执行审计，记录执行中发现的不符合项
- ✦对不符合问题提交不符合项报告，跟踪并验证纠正措施的执行情况
- ✦对项目内不能解决的不符合项问题协调更改
- ✦定期总结质量活动的结果
- ✦制订质量保证的过程改进计划，记录过程数据

8.3. 质量目标

根据相关的质量方针和质量目标，结合本项目特点, 制定项目的总体质量目标：

- 1) 基于需求的测试覆盖率为 100%；
- 2) 软件功能测试用例通过率不低于 95%；
- 3) 每个阶段评审中发现的问题都已经解决或得到适当处理。
- 4) 项目上交时不存在严重及其以上的缺陷。

注：严重问题指导致系统或模块不能正常工作的问题。

结合以往的项目经验和企业的质量相应标准，制定质量标准如下表

表 1：质量计划标准

项 目	具 体 描 述	计 划	实 际
缺陷排除率（缺陷数/页）	需求检查	4	
	系统总体设计检查	2	
	详细设计复核	30	
	详细设计检查	10	
	代码复核	65	
	代码检查	20	
	编译	20	
	单元测试	15	
	系统集成	5	
	系统测试	5	

8. 4. 质量策略

为了保证提交用户的产品是高质量，实施过程中采取的质量保证措施包括：1) 将质量贯彻到日常的项目进展过程中，2) 应该特别注意项目工作产品质量的早期评审工作，无论是质量保证还是质量控制采取的策略都是早期预防和早期排除缺陷。

8. 5. 质量保证活动

质量保证的主要活动包括过程评审和产品审计。过程评审和产品审计的目的是为了确保在项目进展过程的各个阶段和各个方面采取各项措施来保证和提高提交给用户的产品质量。每一次过程评审和产品审计都应填写相应的报告或活动记录。

8. 5. 1. 项目审计

项目审计由组长来进行，检查项目产品是否达到质量目标。
开发人员对项目生存期中创建的工作产品可以有选择性的进行审计，以验证是否符合适当的标准，是否进行了质量检查。表 2 便是质量审计一览表

表 2： 审计产品一览表

项	审计对象	审计阶段	参照的标准
1	软件项目计划	计划结束	企业质量体系
2	软件配置管理计划	计划结束	企业质量体系
3	软件质量保证计划	计划结束	企业质量体系
4	总体设计文档	设计结束	企业质量体系和项目计划
5	详细设计文档	设计结束	企业质量体系和项目计划
6	数据库表和编码规范	设计结束	企业质量体系和项目计划
7	产品代码	每个阶段实施结束	企业质量体系和项目计划
8	测试报告	测试结束	企业质量体系和项目计划
9	测试计划	设计结束	企业质量体系和项目计划
10	用户文档	测试结束	企业质量体系和项目计划

8.5.2 过程评审

项目严格按照组织定义的软件过程进行开发，过程评审的具体依据参照企业的过程规范，保证项目中的所有过程活动都在实施范围内。在每次评审之后，要对评审结果做出明确的决策并形成评审记录。评审可采取文件传阅、评审会等形式。

质量保证人员负责对项目过程进行监督，发现的问题和解决情况在每周的例会上通报，对没有解决的问题进行讨论，对不能解决的问题提交高级管理者处理。

每个周末，进行一次配置管理审核，确认配置管理工作是否正常进行。

根据公司的质量保证体系和本项目的具体特点，确定项目执行过程如下：

- 1) 项目规划过程及产品标准
- 2) 项目跟踪管理过程
- 3) 需求分析过程及产品标准
- 4) 系统设计过程及产品标准
- 5) 详细设计过程及产品标准
- 6) 调试运行过程及产品标准
- 7) 代码走查过程及代码编写标准
- 8) 产品集成测试过程及产品标准
- 9) 开发环境中的执行规则
- 10) 测试环境中的执行规则
- 11) 质量保证过程及其标准
- 12) 配置管理过程及其标准

具体过程执行标准详见企业质量体系 and 项目计划的过程附件。

8.6. 质量控制活动

质量控制活动包括代码走查、单元测试、集成测试、环境测试等，由开发人员负责，详见进度计划。编码人员在编写代码时要进行同步单元测试，单元测试要达到分支覆盖，产品通过单元测试和编码检查后，应提交测试部进行集成测试、系统测试。测试部的测试应达到质量目标要求，软件发布时应达到测试通过准则的要求。

8.7. 记录的收集、维护和保存

项目组应当保留项目执行过程中形成的各类文档、各种记录、各级周报、各级会议记录、对于项目中问题的处理也需要形成记录保存。每周由质量保证人员根据任务清单的审计任务进行审计活动，并收集各活动的过程数据。

9. 软件项目配置管理计划

9.1 引言

包括目的、缩写词和参考资料，具体内容略。

9.2. 组织及职责

配置管理的角色和职责见表 1。

表 1：配置管理角色职责表

角色	人员	职责和工作范围
配置管理者	蒲东齐	制定《配置管理计划》 创建和维护配置库
SCCB 负责人	周宏岳	审批《配置管理计划》 审批重大的变更
SCCB 成员	蒲东齐、周宏岳	审批某些配置项或基线的变更

9.3. 配置管理环境

由于本项目属于中小型项目，工期也不很长，而且项目组人员对 Visual SourceSafe 也比较熟悉，所以采用 Visual SourceSafe 作为配置管理工具。

9.3.1 配置库目录结构

表 2：配置库的目录结构

序号	内容	说明		路径
1	TCM	技术合同管理		\$\prj-manager\TCM
2	RM	需求管理		\$\prj- manager \RM
3	SPP	软件项目规划		\$\prj- manager \SPP
4	SPT0	软件项目跟踪与管理		\$\prj- manager \SPT0
5	SCM	软件配置管理		\$\prj- manager \SCM
6	SQA	软件质量保证		\$\prj- manager \SQA
7	SPE	软件产品	设计	\$\prj- manager \SPE\DESIGN
8		工程	源代码	\$\prj- manager\SPE\SOURCE

9		目标代码	\$\prj- manager \SPE\BUILD
10		测试	\$\prj- manager \SPE\TEST
11		发布	\$\prj-manager\SPE\RELEASE

9.3.2 用户及权限

表 3：配置库的用户权限

类别	人员	权限说明
配置管理者	蒲东齐	负责项目配置管理，拥有所有权限
项目经理	蒲东齐	访问、读，写
质量保证人员	周宏岳	访问、读，写
开发人员	周宏岳	访问、读，写

9.4. 配置管理活动

9.4.1 主要配置项

表 4：配置项列表

类型	主要配置项	标识符	预计正式发表时间
技术合同	《合同》	QTD-School-TCM-Contract-V1.0	2017-12-10
	SOW	QTD-School-TCM-SOW-V1.0	2017-12-11
	《项目计划》	QTD-School-SPP-PP-V1.0	2017-12-15
	《质量保证计划》	QTD-School-SPP-SQA-V1.0	2017-12-18
	《配置管理计划》	QTD-School-SPP-SCM-V1.0	2017-12-19
	《需求规格说明书》	QTD-School-RM-SRS-V1.0	2017-12-20
	用户 DEMO	QTD-School-RM-Demo-V1.0	2017-12-20
	《总体设计说明书》	QTD-School-Design-HL-V1.0	2017-12-22
	《数据库设计》	QTD-School-Design-DB-V1.0	2017-12-22

《详细设计说明书》	QTD-School-Design-LL-V1.0	2017-12-25
《设计术语及规范》	QTD-School-Design-STD-V1.0	2017-12-26
源程序	QTD-School-Code- ModuleName-V1.0	2017-12-30
编码规则	QTD-School-Code-STD-V1.0	2018-1-1
《测试计划》	QTD-School-Test-Plan-V1.0	2018-1-2
《测试用例》	QTD-School-Test-Case-V1.0	2018-1-2
《测试报告》	QTD-School-Test-Report- V1.0	2018-1-4
运行产品	QTD-School-Product-Exe- V1.0	2018-1-5
《验收报告》	QTD-School-Product-Report- V1.0	2018-6-6
《用户手册》	QTD-School-Product-Manual- V1.0	2018-6-6

项目基线

在 Visual SourceSafe 中基线由 LABLE 标志，字母必须为大写。基线管理由项目执行负责人确认、SCCB 授权，由配置管理员执行。

表 5

基线名称/标识符	基线包含的主要配置项	预计建立时间
需求	《需求规格说明书》、用户 DEMO	2017-12-18
总体设计	《总体设计说明书》、《数据库设计》	2017-12-20
项目实现	软件源代码、编码规则	2017-12-20
系统测试	《测试用例》、《测试报告》	2017-12-24

配置项的版本管理

配置项可能包含的分支从逻辑上可以划分成 4 个不同功能的分支：主干分支、私有分支、小组分支、集成分支。让它们分别对应 4 类工作空间。

这四类工作空间（分支）由项目执行负责人统一管理，根据各开发阶段的实际情况定制相应的版本选取规则，来保证开发活动的正常运作。在变更发生时，应及时做好基线的推进。

对配置项的版本管理在不同分支具有不同的策略：

1. 主干分支

系统默认自动建立的物理分支——主干分支（/main），基线均以 LABLE 方式出现在主干分支上。

2. 私有分支

如果多个开发工程师维护一个配置项时建议建立自己的私有分支。配置管理员对其基本不与管理，如个别私有空间上的版本树过于冗余，将对其冗余版本进行限制。

3. 小组分支

如果出现小组共同开发一配置项，该分支可视为项目组内部分组的私有空间，存放代码开发过程中的版本分支，由项目组内部控制。

4. 集成分支

集成测试时在主干分支的特定版本（由 LABLE 标志清晰）上建立集成分支，测试工作在集成分支上完成。

私有分支和小组分支均为可选，必要时建立。

9.4.2 变更管理

变更管理的流程是：

1. 由请求者提交变更请求，SCCB 会召开复审会议对变更请求进行复审，以确定该请求是否为有效请求。典型的变更请求管理有需求变更管理、缺陷追踪等。
2. 配置管理者收到基线修改请求后，在配置库中生成与此配置项相关的波及关系表。
3. 配置管理者将基线波及关系表提交给 SCCB，由 SCCB 确定是否需要修改，如果需要修改，SCCB 应根据波及关系表，确定需要修改的具体文件，并在波及分析表中标志出来。
4. 配置管理者按照出库程序从配置库中取出需要修改的文件。
5. 项目人员将修改后的文件提交给配置管理者。
6. 配置管理者将修改后的配置项按入库程序放入配置库。
7. 配置管理者按 SCCB 标识出的修改文件，由波及关系表生成基线变更记录表，并按入库程序放入配置库。

9.4.3 配置状态统计

利用配置状态统计，可以记录和跟踪配置项的改变。状态统计可用于评估项目风险，在开发过程中跟踪更改，并且提供统计数据以确保所有必需的更改已被执行。为跟踪工作产品基线，配置管理者需手机下列信息：

- 基线类型 ● 工作产品名称
- 配置项名称/标识符 ● 版本号
- 更改日期/时间 ● 更改请求列表
- 需要更改的配置项 ● 当前状态
- 当前状态发生日期

项目组每周提交配置项清单及其当前版本。

配置管理人员每半个月提交变更请求的状态统计。

10 风险计划

1. 风险管理目的

通过对本小组设计开发和生产的系列型体外冲击波碎石机存在的风险进行分析、评价和控制以及生产后信息的评价，确保产品的所有风险均处于可接受的水平。

2. 风险管理活动范围

适用于本团队设计开发和生产的学生信息管理系统在产品实现全过程的风险管理活动的控制，如产品实现的策划、与顾客有关的过程、设计和开发、采购、生产和服务提供、监视和测量装置的控制等风险管理活动的控制。

3. 职责和权限的分配

3.1. 总负责人

- 1) 提供风险管理所需的资源；
- 2) 批准风险管理计划；
- 3) 批准风险管理报告。

3.2. 总工程师

- 1) 全面负责产品技术及质量有关风险管理；
- 2) 参与风险分析和评价。

4. 风险控制措施验证活动

- 1) 验证风险控制措施已在最终设计中得到实施；
- 2) 验证风险控制措施的有效性；
- 3) 所有的对风险控制措施的结果进行验证的活动由技质部负责组织实施；
- 4) 验证活动结束后，由技质部负责记录保持文档。

5. 风险管理活动的评审和验证要求

- 1) 技质部组织参与风险管理的相关部门对风险管理文档进行评审；
- 2) 技质部对风险控制措施的结果进行验证。

6. 风险的可接受性准则

6.1 风险的概率分级

表示符	系数值	风险发生概率定量分析的范围	概率的定性描述	备注
P5	5	$\geq 10^{-2}$	经常	
P4	4	$10^{-3} \sim 10^{-2}$	有时	
P3	3	$10^{-4} \sim 10^{-3}$	偶然	
P2	2	$10^{-5} \sim 10^{-4}$	很少	
P1	1	$< 10^{-5}$	非常少	

6.2 风险的严重度水平

表示符	系数值	风险严重度定量的描述	风险严重度定性的描述	备注
S1	1	可忽略	用户界面不简洁明了	
S2	2	轻度	用户操作繁琐	
S3	3	严重	用户容易误操作	
S4	4	危重的	导致客户端死机或异常卡顿	
S5	5	灾难性的	导致服务器死机	

6.3 风险评价准则

P5	ALARP（合理可行）区	ALARP（合理可行）区	不容许区	不容许区	不容许区
P4	广泛可接受区	ALARP（合理可行）区	ALARP（合理可行）区	不容许区	不容许区
P3	广泛可接受区	ALARP（合理可行）区	ALARP（合理可行）区	ALARP（合理可行）区	不容许区
P2	广泛可接受区	广泛可接受区	ALARP（合理可行）区	ALARP（合理可行）区	ALARP（合理可行）区
P1	广泛可接受区	广泛可接受区	广泛可接受区	广泛可接受区	ALARP（合理可行）区

概率严重度	S1	S2	S3	S4	S5
-------	----	----	----	----	----

6.4 风险评价准则系数值分析

根据上述风险评价准则所构成的 5×5 风险评价矩阵图，确定风险可接受区域。

6.4.1 可接受系数 Ac

$Ac = P_i \times S_i$

6.4.2 广泛可接受区

$Ac \leq 4$

该区域内，风险是可以接受的，并且不需要主动采取风险控制。

6.4.3 ALARP（合理可行）区

$4 < Ac \leq 12$

该区域内，应先考虑接受风险的受益和进一步降低的可行性，然后对风险与受益进行比较，如果受益超过风险，则风险是可接受的；如果受益没有超过风险，则风险是不可接受的。任何风险都应降到可行的最低水平。

6.4.4 不容许区

$Ac > 12$

该区域内，风险如果不能予以降低，则判断为是不容许的。

7. 生产和生产后信息的收集和评审活动

- 1) 生产部提供生产过程与风险有关的相关信息；
- 2) 售后服务部负责对售出产品的质量跟踪，并反馈相关信息（反馈信息的来源包括使用者，服务员，培训人员，事故报告，顾客反馈等）；
- 3) 技质部组织参与风险管理的相关部门对得到的生产和生产后信息进行风险管理；
- 4) 技质部组织参与风险管理的相关部门对风险管理文档进行评审活动。

11 质量度量

表 1

任务名称	规模度量单位	计划	实际
需求规格说明书	文本页数	5	3
总体设计说明书	HLD 页数	2	2
系统测试计划	文本页数	2	2
详细设计说明书	DLD 行数	8	8

单元测试计划	文本页数	2	2
源程序	LOC 行数	10	10
单元测试报告	文本页数	5	5
系统测试报告	文本页数	4	4
总计	文本页数	18	16
	HLD 页数	2	2
	DLD 行数	8	8
	LOC 行数	10	10

12 项目沟通与审核

评审类别	评审周期	评审要点	相关人员
日例会定期 评审（周例会）	每天 17:00 到 17:30 每周五	<ul style="list-style-type: none"> • 不确定主题，随意交流 • 共享经验，避免错误 • 本周工作进度 • 问题及对策 • 资源协调 	全部团队成员
阶段评审	阶段结束	本阶段计划执行情况 质量评审结果 产品审计结果 下阶段修正计划	全部
事件评审	当事件可能影响 计划执行	事件性质和影响范围 事件处理方案的讨论 修改计划的评审时间	全部