2023/11/14

TEAM 11

版本 <2.0>

软件开发计划书

项目名称：网上书店运营系统

的设计与实现

**修订历史记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **制定人** |
| 2023年10月10日 | <1.0> | 初始软件开发计划书 | 孙雁南（项目经理） |
| 2023年11月14日 | <2.0> | 根据课程总体进度更新了项目进度安排，标注了已完成的任务，优化了人力资源分配，重新调整了生命周期部分阶段的迭代里程碑 | 孙雁南（项目经理） |
|  |  |  |  |
|  |  |  |  |

**目录**

[一、项目概述 3](#_Toc5223)

[1.1 项目的目的、规模和目标 3](#_Toc18422)

[1.2 假设与约束 3](#_Toc18375)

[1.3 项目的产品及成果 3](#_Toc8841)

[1.4 软件开发计划的演进 4](#_Toc18071)

[二、项目组织 5](#_Toc13473)

[2.1 组织结构 5](#_Toc13911)

[2.2 角色与职责 6](#_Toc32360)

[三、管理流程 7](#_Toc27788)

[3.2 项目计划 7](#_Toc1352)

[3.2.1     制定项目计划 7](#_Toc12042)

[3.2.2     迭代目标 8](#_Toc15763)

[3.2.3     发布版 10](#_Toc30590)

[3.2.4     项目时间表 10](#_Toc15960)

[3.2.5     项目资源分配 11](#_Toc23660)

[3.2.5.3     人员培训计划 15](#_Toc30964)

[3.2.6     预算 15](#_Toc4863)

[四、项目跟踪 16](#_Toc14884)

[4.1    更新项目进度 16](#_Toc22611)

[4.2    与比较基线做对比 17](#_Toc10399)

# 一、项目概述

## 1.1 项目的目的、规模和目标

本项目主要针对线下实体店传统销售方式中管理不便与效率低的缺点，将电子商务和计算机技术结合起来，开发出管理便捷、效率高的网上书店系统。系统分为两个子系统，其中，前台购物系统是面向顾客的，后台管理系统是面向书店管理员的。本系统能够向顾客用户提供在线浏览图书以及购买等主要功能，向管理员用户提供发货以及上架等主要功能。通过团队合作开发整个系统，以便项目团队根据本计划书开展和检查项目工作。各子系统均采用目前主流的B/S开发架构，将于项目验收后一起发布，不单独发布。交付的产品包括可执行的文件、源代码、技术文档与用户使用手册等。

本系统的开发过程中的主要工作是系统需求分析、系统总体设计、子系统源代码开发、子系统测试、交付组长进行最后的集成、整个系统的测试。关键里程碑是制定项目开发计划书、制定需求设计规格说明书初稿、制定系统设计报告的初稿、进行子系统运行情况的检查与测试、进行系统集成后的运行情况的检查与测试。

## 1.2 假设与约束

为了完成本项目开发，项目团队主要成员角色有项目经理、系统分析员、设计员、开发人员、界面设计师、配置经理、测试经理等。项目所需设备是个人电脑和开发工具。整体开发周期约为11周。

## 1.3 项目的产品及成果

1.3.1程序

软件名称：JLU网上书店

编程语言：Java

功能：实现用户端网上购书和服务端发货、管理库存等主要功能。

2.3.2文件

用户操作手册：本手册详细描述软件的功能、性能和用户界面，使用户对如何使用该软件得到具体的了解,为操作人员提供该软件各种运行情况的有关知识，特别是操作方法的具体细节。

1.3.3非移交产品

本节以表格的形式列出了本项目中创建的非移交产品，以及预期的完成日期和完成成员。

|  |  |  |
| --- | --- | --- |
| 项目工件 | 目标完成日期 | 完成者 |
| 初始软件项目计划 | 2023.10.10 | 项目经理 |
| 软件需求规范 | 2023.10.10 | 系统分析员 |
| 用户界面原型 | 2023.10.10 | 界面设计员 |
| 详细软件项目计划 | 2023.10.25 | 项目经理 |
| 软件测试计划 | 2023.10.25 | 测试经理 |
| 配置管理计划 | 2023.10.25 | 配置经理 |
| 设计文档 | 2023.11.22 | 设计员 |
| 软件测试设计 | 2023.12.6 | 测试经理 |
| 测试报告 | 2023.12.6 | 测试经理 |
| 配置库管理文件 | 2023.12.13 | 配置经理 |
| 源程序 | 2023.12.13 | 开发人员 |

## 1.4 软件开发计划的演进

本计划书以表格的形式列出了软件开发计划的修订版本，以及在计划外修订与重新发行此计划需符合的标准。将软件开发计划的书写分解为多次迭代，每次迭代后及时汇报、评审和调整，逐步完善。

修订标准如下：

项目开发计划书在第三周周末前经由小组讨论、项目经理汇总整合形成初稿，第四周以后根据项目的进展可以对其进行修改，需要由组员提出修改意见，在全体会上讨论通过，并由项目经理整理修改意见并作出相应的修改和记录，其余组员同步获得更新稿。

# 二、项目组织

## 2.1 组织结构

针对项目需求与团队规模，对团队成员采用了职能型组织管理方式，按职能来组织部门分工，每个部门负责特定的职能。项目经理有足够的权力控制项目资源。项目经理权力充分，团队各成员职责清晰，分工明确，互不干扰。

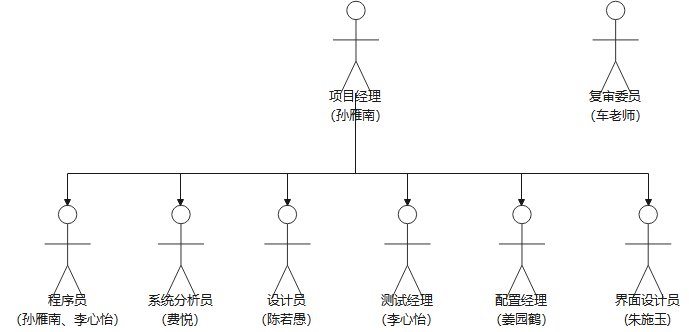


图3.1 项目团队组织结构图

## 2.2 角色与职责

在软件开发项目中，不同角色扮演着关键的职责，协同合作以推动项目的顺利进行。以下是团队成员的开发角色及其职责：

|  |  |  |
| --- | --- | --- |
| 成员 | 角色 | 职责 |
| 孙雁南 | 项目经理 | 日常项目管理、组织制定项目计划、负责软件的交付工作。 |
| 费悦 | 系统分析员 | 整理需求分析并撰写软件需求规范、维护并及时修改和发布已更新技术文档。作为程序员还要参与软件设计与代码开发。 |
| 陈若愚 | 设计员 | 组织开发分析设计模型：子系统、类、接口、用例实现，并撰写软件设计报告。 |
| 李心怡 | 测试经理 | 主要负责软件代码测试和用户测试、并撰写软件测试计划、测试设计和测试报告。 |
| 姜园鹤 | 配置经理 | 主要负责源代码的版本管理工作，负责撰写配置管理计划和配置库的管理工作。 |
| 朱施玉 | 界面设计员 | 主要负责界面美工，开发用户界面原型和编写用户手册。 |
| 孙雁南  李心怡 | 程序员 | 主要负责软件设计和编写代码。 |

# 三、管理流程

### 3.2 项目计划

## 3.2.1     制定项目计划

本项目采用迭代式开发，通过将开发过程划分为多个迭代，每个迭代都是一个完整的开发循环，在每个迭代中逐步完善需求和功能。项目需求可以在每个迭代中逐步完善，适应变化。每个迭代都可以产生可用的部分产品，并通过快速原型、增量开发实现。

下表所列的是计划各个阶段时间段和迭代次数：

|  |  |  |
| --- | --- | --- |
| 时间 | 项目开发阶段 | 迭代次数 |
| 2023年9月27日——2023年10月11日 | 初始阶段 | 1 |
| 2023年10月12日——2023年11月7日 | 细化阶段 | 2 |
| 2023年11月9日——2023年12月4日 | 构建阶段 | 2 |
| 2023年12月5日——2023年12月12日 | 移交阶段 | 2 |

确定主要里程碑及其成功标准：

我们开发项目整体有四个主要里程碑，分别是初始阶段的目标里程碑、细化阶段的构架里程碑、构造阶段的初试运作能力里程碑和交付阶段的产品发布里程碑。根据RUP生命周期对每一个阶段以及标志阶段完成的主要里程碑都进行了说明。

|  |  |  |
| --- | --- | --- |
| 阶段 | 说明 | 里程碑 |
| 初始阶段 | 将确定产品的需求并建立网上书店运营系统的商业理由。不仅要编制主要用例，同时也要制订项目开发计划。 | 处在阶段结束时的商业理由复审里程碑是项目启动的决策标志。 |
| 细化阶段 | 分析需求并开发构架原型。阶段结束时，将完成对 1.0 发布版所选的所有用例的分析设计。除此在外，还将完成对 2.0 发布版高风险用例的分析和设计。构架原型将测试 1.0 发布版所需构架的可行性和性能。 | 构架原型里程碑标志着细化阶段的结束。该原型标志着对组成 R1.0 发布版的主要构架构件的验证和确认。 |
| 构造阶段 | 将对余留的用例进行分析和设计。将开发并分发发布版 1.0 的 Beta 版本以供评估。并将完成支持 R1.0 和 R2.0 发布版的实施和测试活动。 | R2.0 操作性能里程碑标志着构造阶段的结束。软件的 2.0 发布版已经准备包装。 |
| 发布阶段 | 将做好发行 R1.0 和 R2.0 发布版的准备。提供包括用户培训在内的确保安装顺利所需要的各种支持。 | R2.0 发布版里程碑标志着产品化阶段的结束。此时在前景文档中确定的所有功能都已经安装，并且对于用户是可以使用的。 |

## 3.2.2     迭代目标

在整个项目的开发过程中，每一个生命周期阶段都由若干开发迭代组成。迭代过程中将对系统的子集，也就是每一个子系统模块进行迭代开发。一般说来，这些迭代将：

降低技术风险；

提供一个工作系统的早期版本；

在计划每一个发布版的功能时具有非常充分的灵活性；

在一个迭代周期之内，能够有效地处理范围变更。

下表在说明了迭代以及相关的里程碑和所处理的风险。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 阶段 | 迭代 | 说明 | 相关里程碑 | 风险处理 |
| 初始阶段 |  | 确定业务模型、产品需求、项目计划和商业理由。 | 商业理由复审 | 预先阐明用户需求。制订实际的项目计划和范围。从商业角度确定项目的可行性。 |
| 细化阶段 | #1 | 完成所有用例的分析设计。开发的构架原型。 | 初步构架原型 | 阐明了构架问题，降低技术风险。 |
| #2 | 完成所有高风险用例的分析设计。 | 构架原型 |
| 构造阶段 | #1 | 实施和测试用例以提供Alpha版本。 | Alpha版本 | 在Alpha版本中实施了所有从用户和构架角度提出的关键功能。 |
| #2 | 实施和测试余留用例，修复Alpha版本中的缺陷，并结合Alpha版本中得到的反馈开发Beta版本。 | Beta版本 | 由用户群对软件进行充分复审。 |
| 发布阶段 | #1 | Debug、测试、发布RC版本 | RC版本发步 | 发布分为两个阶段进行可将缺陷降到最低。 |
| #2 | Debug、测试、发布GA版本 | GA版本发布 | 发布分为两个阶段进行更方便用户过渡。 |

## 

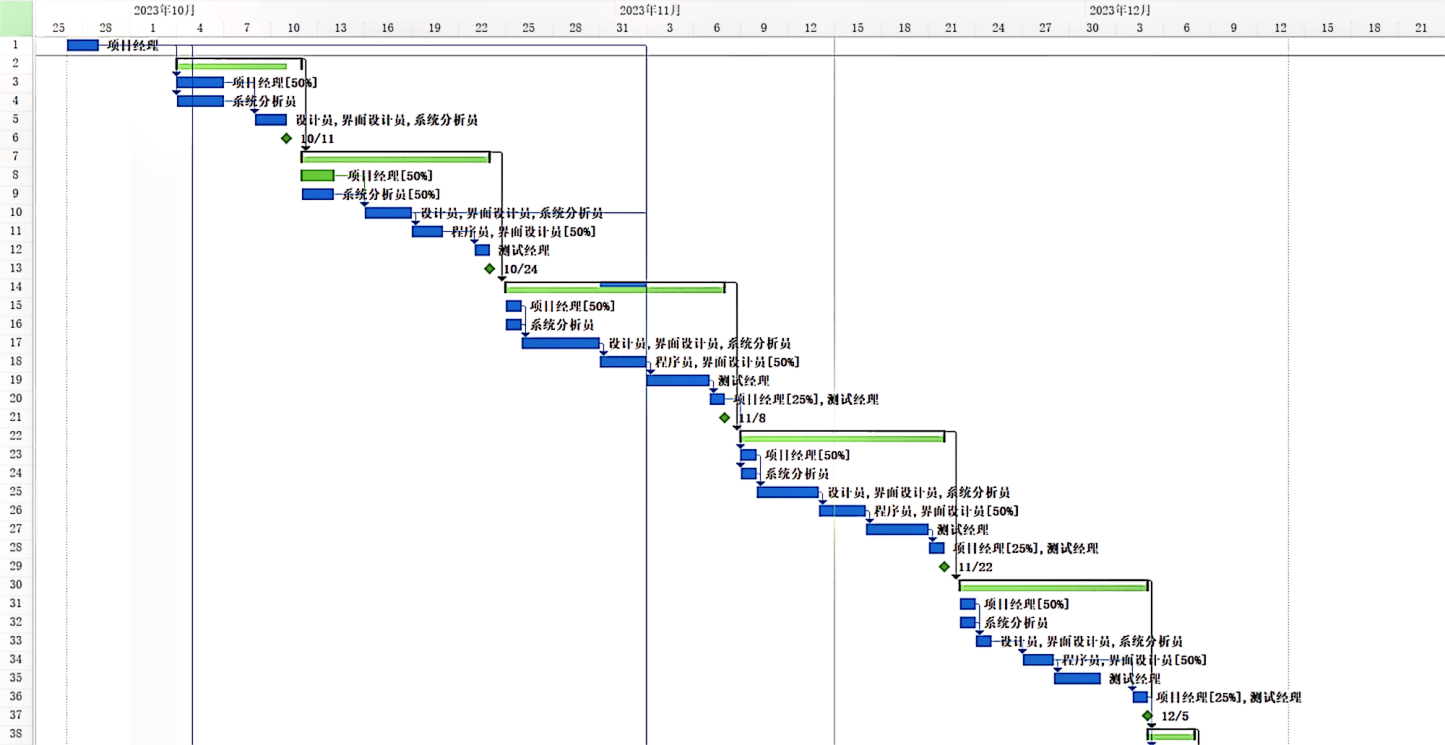
## 3.2.3     发布版

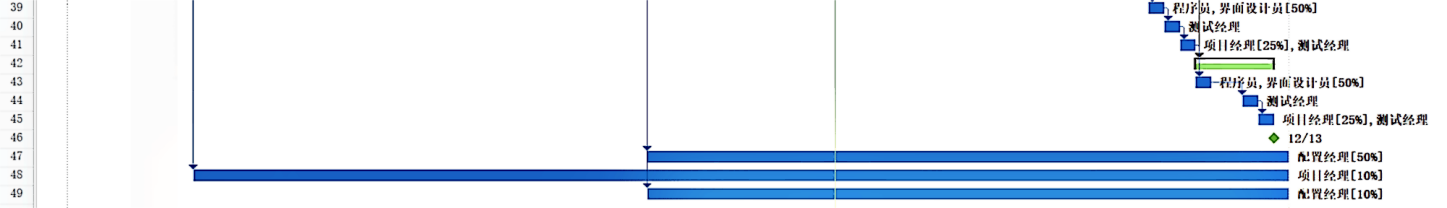
简要说明每个软件发布版，是否是演示版、Beta 版等。

|  |  |
| --- | --- |
| 版本 | 要求 |
| Alpha版本 | 进行内部测试，发现并修复软件中的问题，同时也可以提供一个早期的版本，供开发团队内部或特定的测试用户使用，以便获取反馈。 |
| Beta版本 | 对Alpha版本进行测试和改进，完善全部用例，如处理图书退货、统计各类别销量等功能，并对用户界面进行修改完善 |
| RC版本 | 较少明显的bug，较为成熟 |
| Release版本 | 基本无用户能明显察觉的bug，性能达到预期的90% |

## 3.2.4     项目时间表

具体开发进度安排的甘特图如下：

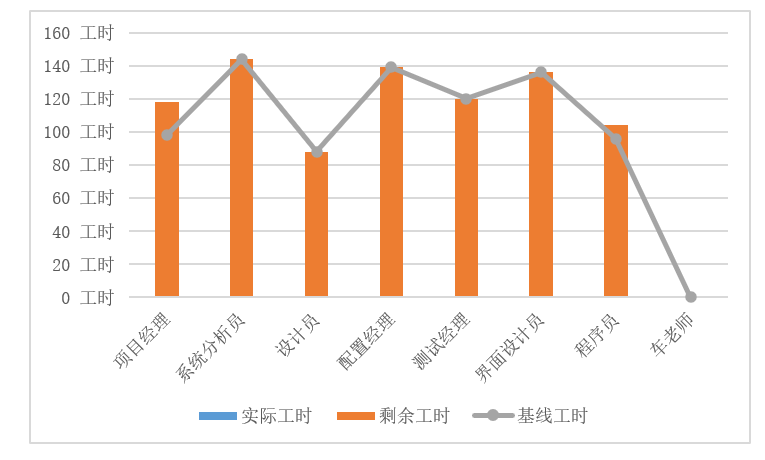




## 3.2.5     项目资源分配

在项目管理中，资源分配是确保项目按计划进行的关键环节。应当尽量避免不断将资源从一个项目转移到另一个项目，以提高员工效率，避免超负荷。

本项目人力资源的工时状态柱状图如下：



以下表格为成员角色在各任务中的具体资源分配情况：

|  |  |  |  |
| --- | --- | --- | --- |
| 资源名称 | 总工时 | 任务名称 | 任务资源 |
| 项目经理 | 118 工时 | 制定项目计划 | 16 工时 |
| 初始 业务建模 | 12 工时 |
| 细化#1 业务建模 | 8 工时 |
| 细化#2 业务建模 | 4 工时 |
| 细化#2 部署 | 2 工时 |
| 构造#1 业务建模 | 4 工时 |
| 构造#1 部署 | 2 工时 |
| 构造#2 业务建模 | 4 工时 |
| 构造#2 部署 | 2 工时 |
| 移交#1 部署 | 2 工时 |
| 移交#2 部署 | 2 工时 |
| 项目管理 | 60 工时 |
| 系统分析员 | 144 工时 | 初始 需求 | 24 工时 |
| 初始 分析与设计 | 16 工时 |
| 细化#1 需求 | 8 工时 |
| 细化#1 分析与设计 | 24 工时 |
| 细化#2 需求 | 8 工时 |
| 细化#2 分析与设计 | 24 工时 |
| 构造#1 需求 | 8 工时 |
| 构造#1 分析与设计 | 16 工时 |
| 构造#2 需求 | 8 工时 |
| 构造#2 分析与设计 | 8 工时 |
| 设计员 | 88 工时 | 初始 分析与设计 | 16 工时 |
| 细化#1 分析与设计 | 24 工时 |
| 细化#2 分析与设计 | 24 工时 |
| 构造#1 分析与设计 | 16 工时 |
| 构造#2 分析与设计 | 8 工时 |
| 配置经理 | 139.2 工时 | 配置与变更管理 | 116 工时 |
| 环境 | 23.2 工时 |
| 测试经理 | 120 工时 | 细化#1 测试 | 8 工时 |
| 细化#2 测试 | 16 工时 |
| 细化#2 部署 | 8 工时 |
| 构造#1 测试 | 16 工时 |
| 构造#1 部署 | 8 工时 |
| 构造#2 测试 | 24 工时 |
| 构造#2 部署 | 8 工时 |
| 移交#1 测试 | 8 工时 |
| 移交#1 部署 | 8 工时 |
| 移交#2 测试 | 8 工时 |
| 移交#2 部署 | 8 工时 |
| 界面设计员 | 136 工时 | 初始 分析与设计 | 16 工时 |
| 细化#1 分析与设计 | 24 工时 |
| 细化#1 实现 | 8 工时 |
| 细化#2 分析与设计 | 24 工时 |
| 细化#2 实现 | 12 工时 |
| 构造#1 分析与设计 | 16 工时 |
| 构造#1 实现 | 12 工时 |
| 构造#2 分析与设计 | 8 工时 |
| 构造#2 实现 | 8 工时 |
| 移交#1 实现 | 4 工时 |
| 移交#2 实现 | 4 工时 |
| 程序员 | 104 工时 | 细化#1 实现 | 24 工时 |
| 细化#2 实现 | 24 工时 |
| 构造#1 实现 | 24 工时 |
| 构造#2 实现 | 16 工时 |
| 移交#1 实现 | 8 工时 |
| 移交#2 实现 | 8 工时 |
| 车老师 | 48 工时 | 目标需求评审 | 8 工时 |
| 初步构架评审 | 8 工时 |
| 总体构架评审 | 8 工时 |
| 初始运作能力评审 | 8 工时 |
| 集成运作能力评审 | 8 工时 |
| 产品交付评审 | 8 工时 |

## 3.2.5.3     人员培训计划

1、项目管理培训：项目经理

培训内容：学习MS Project项目计划和计划工具的使用，培养团队管理和领导力，进行实际案例分析和项目模拟训练。

目标时间：于第3周完成。

2、测试与配置培训：测试经理、配置经理。

培训内容：熟悉单元测试和集成测试的基本流程和基本概念，以及如何进行配置项测试的知识和技能，以确保软件产品的质量和稳定性。

目标时间：于第4周完成。

3、美工培训：系统设计员、界面设计员、开发人员

培训内容：学习图形设计基础知识，熟悉即时设计等设计软件的使用，使用HTML和CSS进行基本网页布局和设计。

目标时间：于第6周完成。

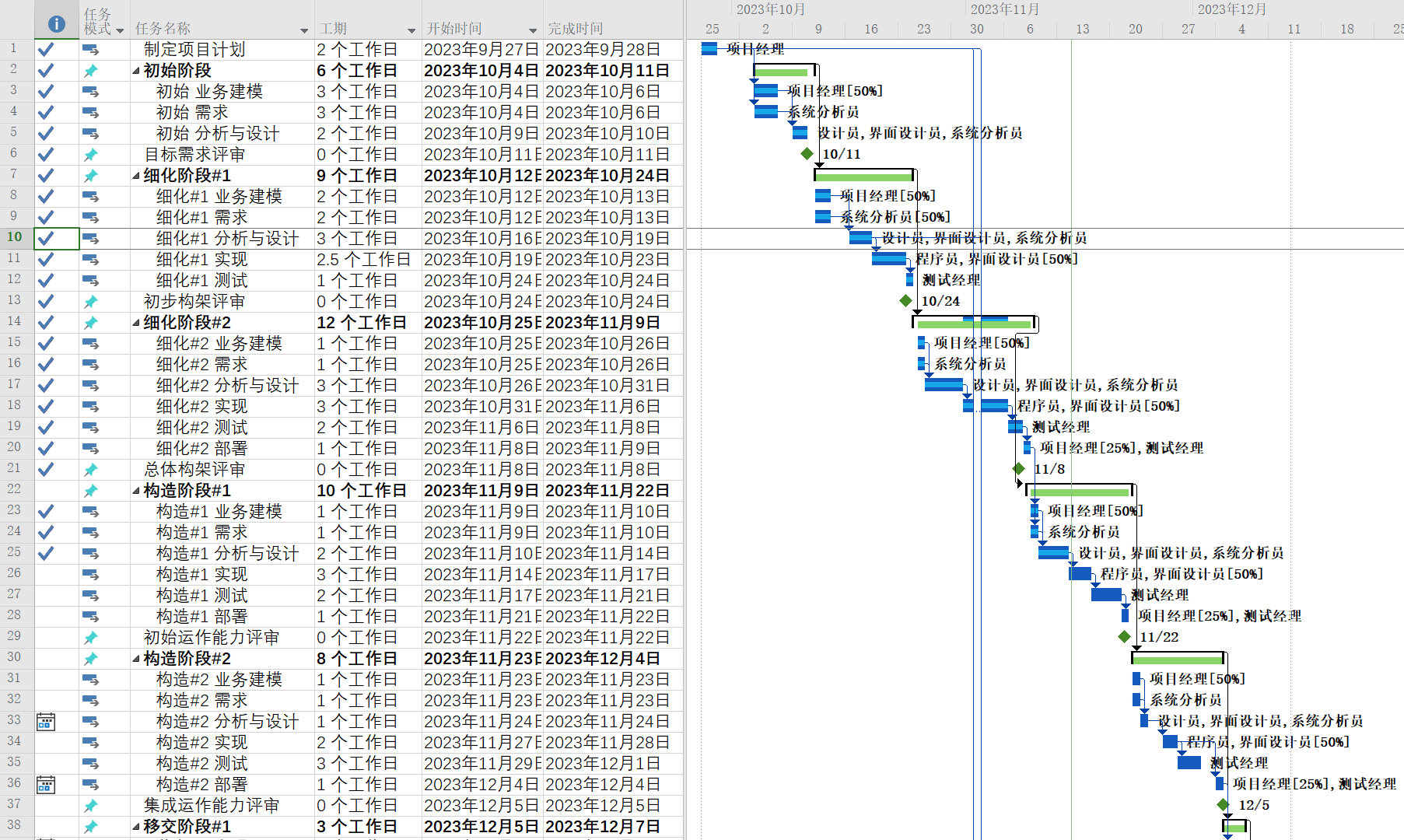
## 3.2.6     预算

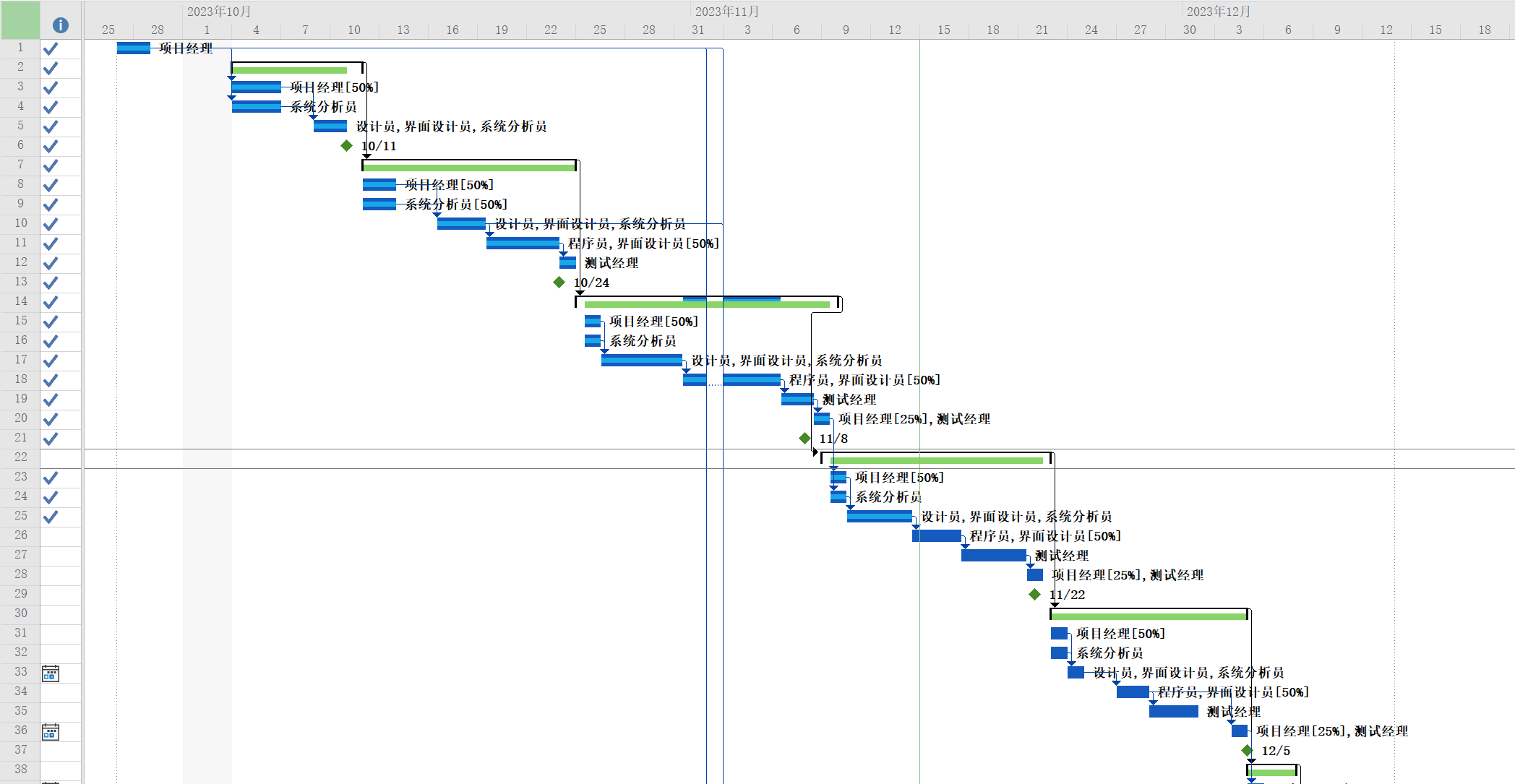
统一的开发不涉及任何经济的预算。资源分配为各自使用自己的电脑。

# 四、项目跟踪

## 4.1    更新项目进度

根据新的课程总体进度更新了后续的项目进度安排，跟踪记录了已完成的任务，优化了人力资源分配，重新调整了生命周期部分阶段的迭代里程碑，设立新的基线。





## 4.2    与比较基线做对比

截至11月15日，项目开发大致按照初始软件开发计划顺利进行。将更新后的项目进度与基线比较，细化阶段的第一次迭代中，分析与设计任务推迟了1个工作日，实现任务缩减至2.5个工作日。对于构架里程碑及后续任务的确立影响不大。细化阶段的第二次迭代中，实现工作于11月1日暂停，推迟了1个工作日完成。

