



软件工程课程实践说明

北京大学软件工程课程
2023年9月



Peking
University

- 软件工程课程实践的目的在于通过实验加深学生对软件工程课程基本理论、基本知识的理解，提高分析和解决软件项目问题的能力；
- 使学生熟悉软件工程的规范、项目管理和团队协作开发，为后续的学习、研究和软件开发工作打下良好的基础。
- 软件工程课程实践要求学生组成项目组，以项目组方式选择感兴趣的实验课题，进行软件开发计划、软件需求分析、系统设计、编程实现、软件测试、软件开发总结等软件工程实践活动，并以小组方式进行课程实践的课堂报告，并且按照要求形成详细的课程实践的文档报告。





- 课程实践建议分为五个阶段，具体要求将在后面详细介绍
- 需提交的文档建议命名为“第x阶段-文档名”的格式，并单独提交到项目仓库的文档文件夹中，或建立另一个仓库用于提交文档，文档标准见“文档标准”文件夹
- 在“各阶段文档示例”文件夹中可以查看以往课程实践项目的文档示例



□ 第一阶段

➤ 内容：

- 项目选题，项目开发计划、需求捕获与描述（列出核心需求）
- 确定技术选型，展示初步学习成果
- 进行初步UI原型设计

➤ 建议时间：3周

➤ 需完成内容：

- 项目选题，项目开发计划、需求捕获与描述
- 技术选型：包含Demo展示，以及一个类似开源项目介绍或自己对本项目的规划
- GitHub 项目仓库的建立，预计的小组分工



□ 第一阶段

➤ 提交文档内容：

- 项目开发计划（见“文档标准”文件夹）
- 项目开发计划甘特图（例如：MS Project / Gantt Project）
- 需求捕获（use case图及文档 / 用户故事，考虑可能的需求变化）
- 软件需求说明书（见“文档标准”文件夹）
- 系统的UI设计稿
- 课堂报告PPT

➤ Demo要求：

- 对于开发 Web 应用的，要求在 /test/hello 路径下，显示 hello world 或者展示部分应用功能
- 对于开发 App 的
 - 要求展示 App 的一个页面，该页面需要包含 hello world 字样，或者展示 App 部分功能
 - 要求 App 后端在 /test/hello 路径下，显示 hello world 字样 或者其他内容



□ 第一阶段

➤ 项目介绍要求

- 如果选择介绍开源项目：主要按照模块介绍项目怎么运行
- 如果选择介绍本项目：按照模块对本项目进行功能上的设计规划

➤ 其他要求

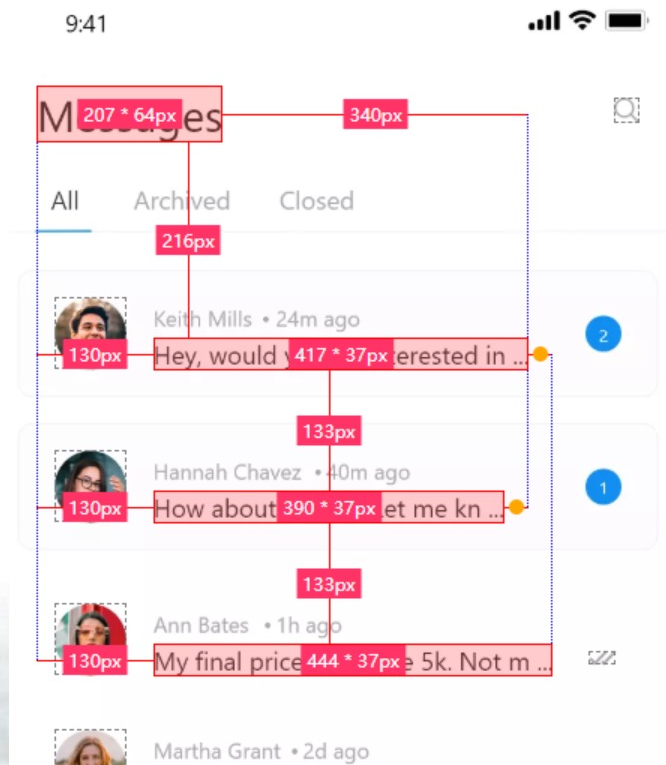
- 组内每位成员都要建立 GitHub 账户
- 由其中一位成员建立一个(或多个, 依照项目需要) GitHub 项目仓库(项目名称自拟)
- 建立仓库的同学将组内成员加入仓库的 Collaborators (每位同学至少属于某一个仓库的 Collaborators)



□标注图

- 对界面元素的具体规格进行标注，保证开发的顺利进行。

- 例：



□切图

- 将设计稿制作成开发中用到的各种图像资源。
- 可能包括：界面背景图、按钮背景图、应用图标等等。

- 例：



□ 第二阶段

- 内容：应用面向对象方法进行核心功能的分析和设计，项目进度，git的应用情况
- 建议时间：5周
- 需完成内容：
 - 面向对象分析（OOA）
 - 用况图、类图、顺序图、状态图、包图等
 - 面向对象设计（OOD）
 - 人机界面、控制驱动、数据管理等
 - 项目进度展示
 - 甘特图执行情况、backlog
 - 要求必须实现至少一个核心功能并进行展示
 - GitHub项目状态展示
 - 主要包含提交(commit)、分支(branch)、贡献(contribution)、kanban等情况



□ 第二阶段

➤ 提交文档内容:

- 项目执行情况甘特图、项目Backlog, 项目Kanban
- 面向对象分析 (OOA) 图及文档 (核心功能)
 - 用况图, 类图, 顺序图, 状态图, 包图
 - 参考“文档标准”文件夹的面向对象分析设计文档
- 面向对象设计 (OOD) 图及文档 (核心功能)
 - 人机界面, 控制驱动, 数据管理
 - 参考“文档标准”文件夹的面向对象分析设计文档
- 课堂报告PPT



□ Backlog和kanban说明

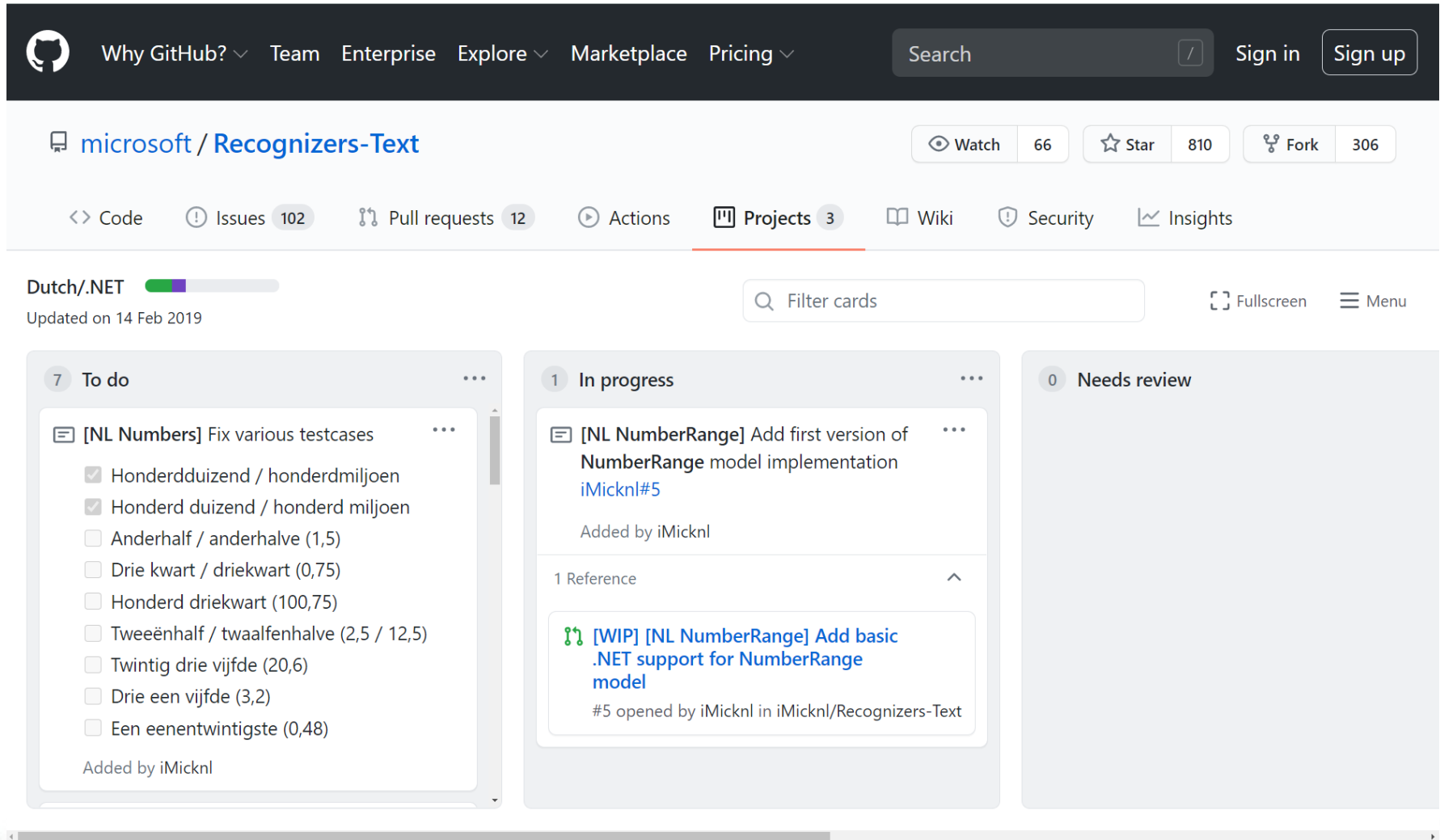
- backlog是一个具有优先级的需求列表，并对每个需求进行了粗略的估算，其中包括可预知的所有任务
- Kanban是一种对敏捷软件开发流程的管理方法，将所有任务和其进展放置于Kanban上，便于对项目进度进行管理



产品 BACKLOG (示例)					
ID	Name	Imp	Est	How to demo	Notes
1	存款	30	5	登录, 打开存款界面, 存入 10 欧元, 转到我的账户余额界面, 检查我的余额增加了 10 欧元。	需要 UML 顺序图。目前不需要考虑加密的问题。
2	查看自己的交易明细	10	8	登录, 点击“交易”, 存入一笔款项。返回交易页面, 看到新的存款显示在页面上。	使用分页技术避免大规模的数据库查询。和查看用户列表的设计相似。



GitHub Kanban示例



The screenshot shows the GitHub interface for the `microsoft/Recognizers-Text` repository. The top navigation bar includes links for Why GitHub?, Team, Enterprise, Explore, Marketplace, and Pricing, along with a search bar and Sign in / Sign up buttons. The repository name is displayed as `microsoft / Recognizers-Text`, with statistics for Watch (66), Star (810), Fork (306), and a list of tabs: Code, Issues (102), Pull requests (12), Actions, Projects (3), Wiki, Security, and Insights. The Kanban board is titled `Dutch/.NET` and was updated on 14 Feb 2019. It features three columns: **To do** (7 cards), **In progress** (1 card), and **Needs review** (0 cards). The **To do** column contains a card titled `[NL Numbers] Fix various testcases` with a checklist of Dutch number phrases and their numerical equivalents. The **In progress** column contains a card titled `[NL NumberRange] Add first version of NumberRange model implementation` by `iMicknl#5`, which includes a reference to a pull request: `[WIP] [NL NumberRange] Add basic .NET support for NumberRange model`. The **Needs review** column is currently empty.

Why GitHub? Team Enterprise Explore Marketplace Pricing

Search Sign in Sign up

microsoft / Recognizers-Text

Watch 66 Star 810 Fork 306

<> Code ! Issues 102 Pull requests 12 Actions Projects 3 Wiki Security Insights

Dutch/.NET Updated on 14 Feb 2019

Filter cards Fullscreen Menu

7 To do

- [NL Numbers] Fix various testcases
 - ☒ Honderdduizend / honderdmiljoen
 - ☒ Honderd duizend / honderd miljoen
 - ☐ Anderhalf / anderhalve (1,5)
 - ☐ Drie kwart / driekwart (0,75)
 - ☐ Honderd driekwart (100,75)
 - ☐ Tweeënhalf / twaalfenhalve (2,5 / 12,5)
 - ☐ Twintig drie vijfde (20,6)
 - ☐ Drie een vijfde (3,2)
 - ☐ Een eenentwintigste (0,48)

Added by iMicknl

1 In progress

- [NL NumberRange] Add first version of NumberRange model implementation
 - Added by iMicknl
 - 1 Reference
 - [WIP] [NL NumberRange] Add basic .NET support for NumberRange model
 - #5 opened by iMicknl in iMicknl/Recognizers-Text

0 Needs review



□ 第三阶段

- 内容：第一次迭代
- 建议时间：1周
- 需完成内容：
 - 项目进度展示（至少实现60%的核心功能）
 - 甘特图执行情况、backlog
 - 项目代码风格规范
 - GitHub项目状态展示
 - 主要包含提交(commit)、分支(branch)、贡献(contribution)、Kanban等情况
 - 项目Demo
- 文档提交内容
 - 项目甘特图、backlog执行情况，项目Kanban，代码风格简述
 - 课堂报告PPT



□ 第三阶段

➤ 项目Demo要求

- 简要介绍已实现的核心功能（至少实现60%的核心功能）
- 针对描述的每个核心功能，展示自己的项目Demo、对功能进行演示
 - 可以现场演示
 - 也可录制演示视频
- 介绍核心功能的开发实现情况



□ 第四阶段

- 内容：第二次迭代，核心功能开发与测试
- 建议时间：3周
- 需完成内容：
 - 测试分析
 - 项目进度展示（至少实现80%核心功能）
 - 甘特图执行情况、backlog
 - GitHub项目状态展示
 - 主要包含提交(commit)、分支(branch)、贡献(contribution)、Kanban等情况
 - 新增或改动的设计分析介绍
 - OOA和OOD的增加和更新
 - 项目Demo



□ 第四阶段

➤ 文档提交内容：

- 项目执行情况甘特图、项目backlog、项目Kanban
- 测试分析报告（见“文档标准”文件夹）
- 面向对象分析（OOA）图及文档（增加与更新）
 - 用况图，类图，顺序图，状态图，包图
 - 参考“文档标准”文件夹的面向对象分析设计文档
- 面向对象设计（OOD）图及文档（增加与更新）
 - 人机界面，控制驱动，数据管理
 - 参考“文档标准”文件夹的面向对象分析设计文档
- 课堂报告PPT



□ 第五阶段

- 内容：第三次迭代，测试、总体成果展示、项目总结
- 建议时间：1周
- 需完成内容：
 - 项目介绍
 - 项目最终成果演示
 - 测试分析报告（增加与更新）
 - 项目进度展示
 - 甘特图执行情况、backlog
 - GitHub项目状态展示
 - 主要包含提交(commit)、分支(branch)、贡献(contribution)、Kanban等情况



□ 第五阶段

➤ 文档提交内容：

- 项目执行情况甘特图、项目backlog、项目Kanban
- 项目开发安装包（包含部署说明等文档）
- 测试分析报告（增加与更新）
- 项目开发总结报告（见“文档标准”文件夹）
- 课堂报告PPT



□ 结束时需完成的所有材料

➤ 内容：

- 项目执行情况甘特图
- 实践过程中所编写的所有图及文档的最终版本
- 实践报告的所有PPT
- 软件可运行或部署版本（包含部署说明等文档）
 - “**Web应用**” 提交部署所需的所有文件，如war包和数据库设计
 - “**移动应用**” 提交部署所需的所有文件，如.apk和.war等
- 小组成员的分工说明以及课程感悟建议
- GitHub仓库链接
- 软件使用说明书（可选）



□ 文档编写

- 遵循国家标准，适当剪裁，形成以下软件工程文档标准，以指导实验（可从“文档标准”文件夹中获取）：
 - 项目开发计划
 - 软件需求说明书
 - 测试分析报告
 - 项目开发总结报告
- 其余文档的格式可根据需要自行规定，但应参考上述标准文档的样式，如目录等



□工具软件

- Microsoft Project、UPGantt、Gantt Project
 - 辅助完成项目开发计划和项目开发总结用户手册
- Microsoft Visio、Visual Paradigm Community Edition、Process On
 - 画图工具，辅助完成文档中标准化图示的制作
- Sketch、Figma、墨刀
 - UI设计工具
- Postman
 - API接口测试工具
- Git
 - 分布式版本控制工具
- GitHub
 - 提供了看板、甘特图、燃尽图等
- 注：以上为较为通用的软件，基于所选技术不同，会有不同的工具



□ 技术和框架

➤ 前端

- **网页前端**: HTML+CSS+Javascript; Ajax, ES6, Webpack; Vue, React, Angular
- **Android**: Kotlin, 五大组件, Retrofit, Material Design, Jetpack
- **IOS**: 四层结构, Swift
- **微信小程序**: WXML, WXSS, 微信提供的API接口
- **跨平台**: Flutter, 声明式框架, React-native, Xamarin

➤ 后端

- **Java**: Spring, SpringMVC, MyBatis, Maven; SpringBoot
- **Python**: Django, Flask, Tornado
- **Nodejs**: Express.js, Koajs, NestJS

➤ 数据库

- MySQL, PostgreSQL, MongoDB, Redis



□ 任务类视图

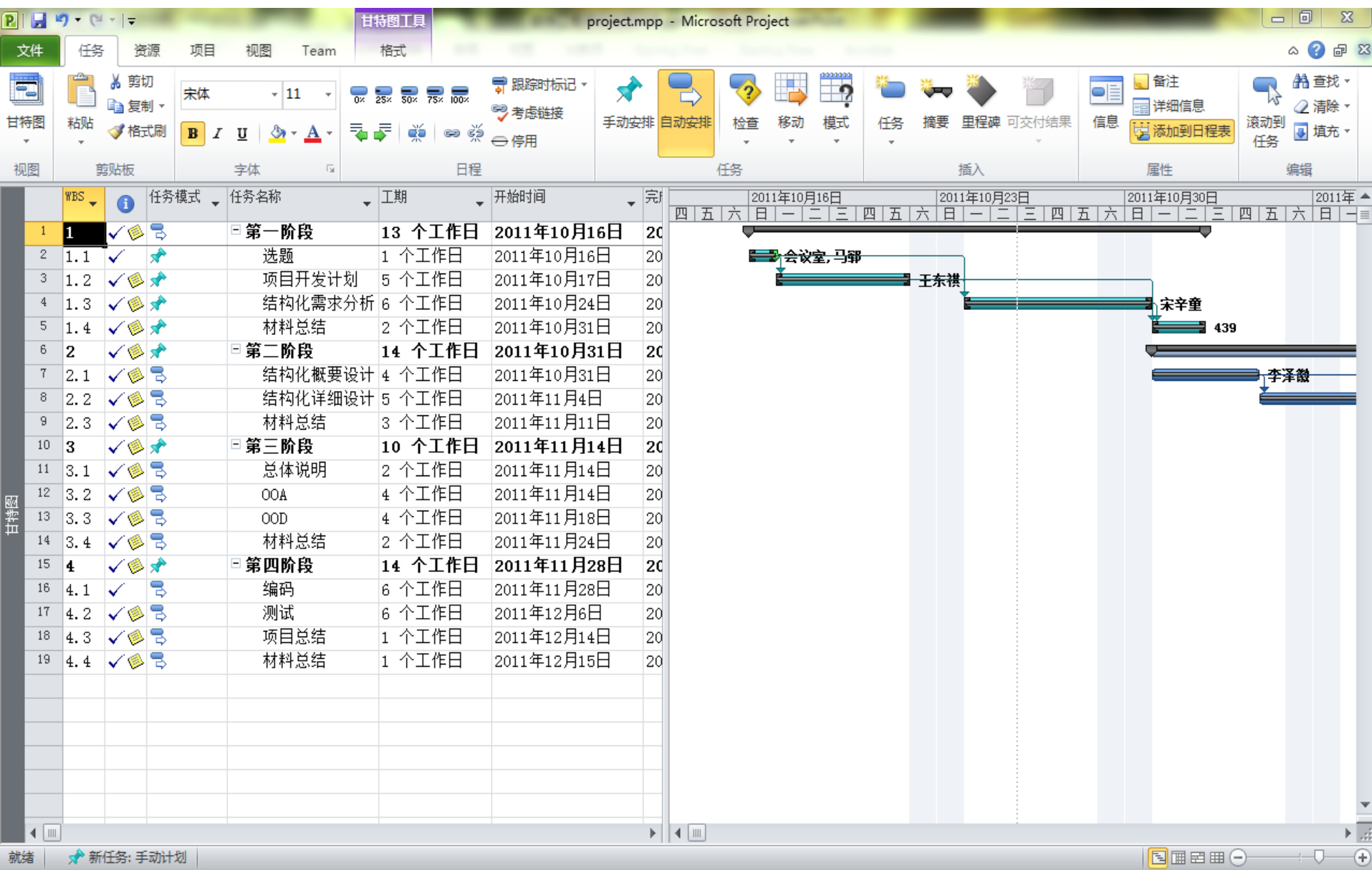
- “甘特图” 视图
- “网络图” 视图
- “日历” 视图
- “任务分配状况” 视图

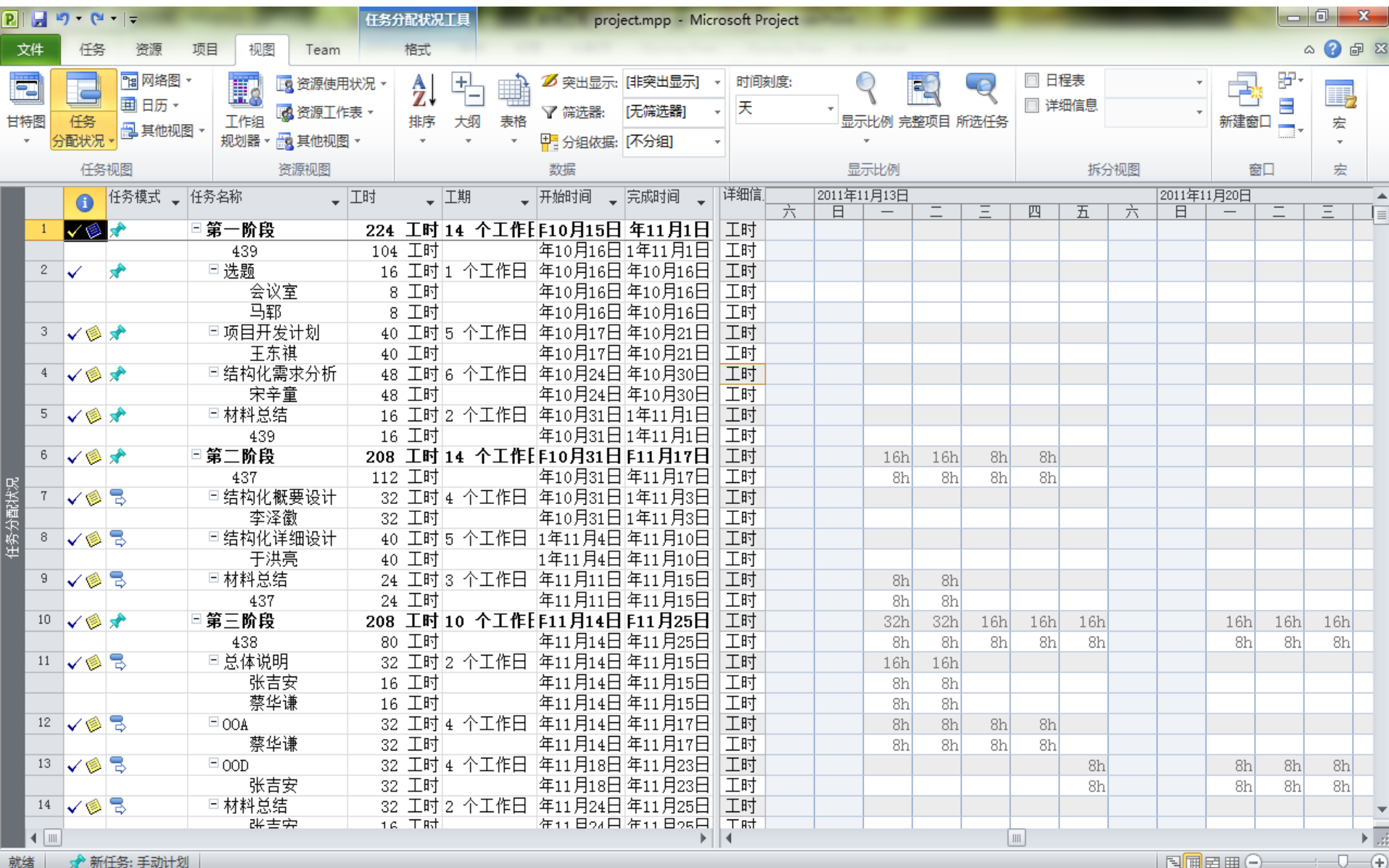
□ 资源类视图

- “资源工作表” 视图
- “资源图表” 视图
- “资源使用状况” 视图



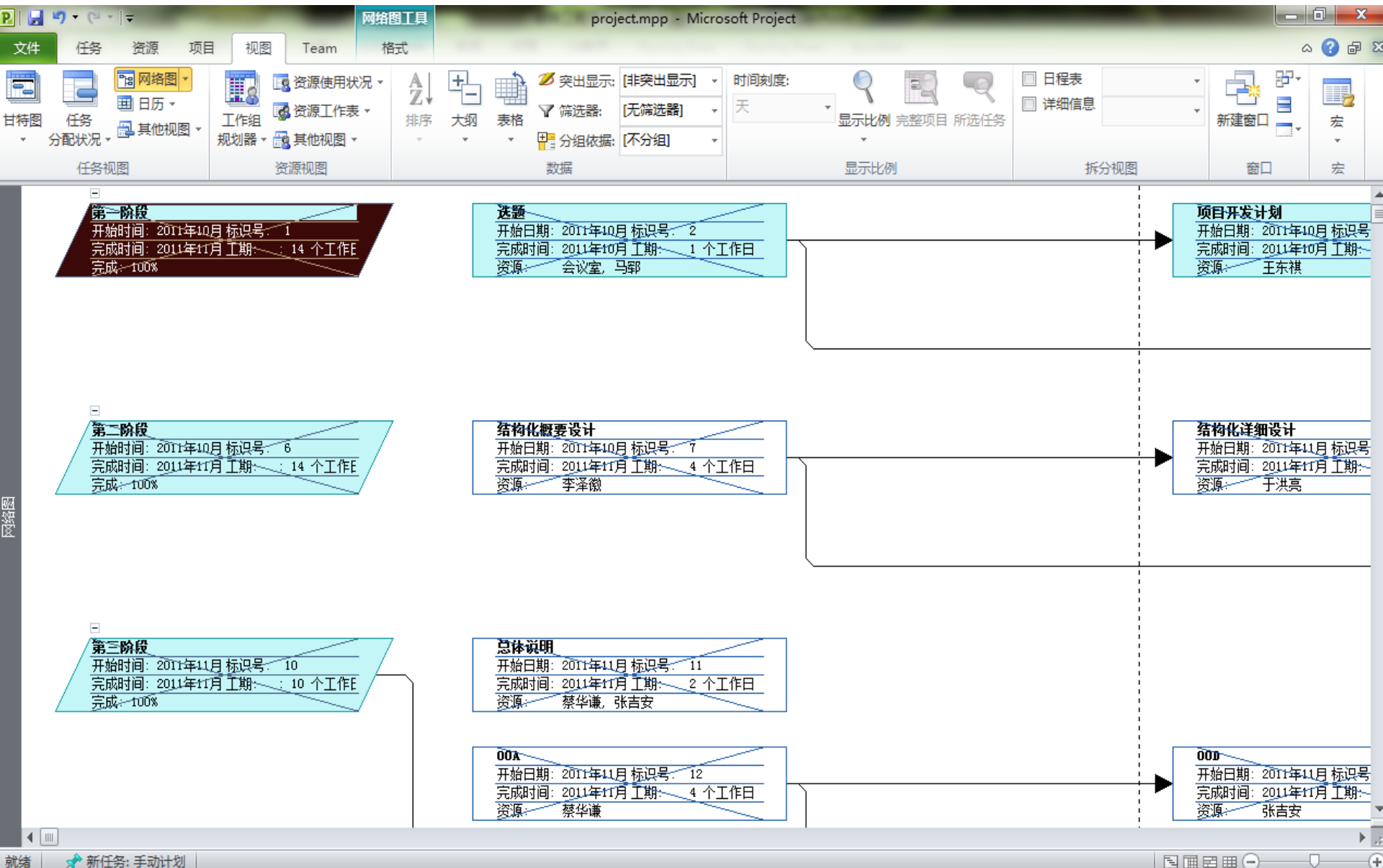
“甘特图”视图







“网络图”视图



“资源工作表”视图

project.mpp - Microsoft Project

文件 任务 资源 项目 视图 Team 格式

资源工作表工具

网络图 资源使用状况 资源工作表 工作组规划器 其他视图

排序 大纲 表格 突出显示: [非突出显示] 时间刻度: 天 显示比例: 完整项目 所选任务

筛选器: [无筛选器] 分组依据: [不分组]

日程表 详细信息

新建窗口 宏

	资源名称	类型	材料标签	缩写	组	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	代码	添加新列
1	会议室	工时		会		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	24 小时		
2		工时		马		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
3	439	工时		4		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
4	438	工时		4		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
5	437	工时		4		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
6		工时		王		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
7		工时		宋		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
8		工时		李		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
9		工时		于		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
10		工时		张		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
11		工时		蔡		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
12		工时		赵		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		
13		工时		关		100%	¥0.00/工时	1.00/工时	¥0.00	按比例	标准		

就绪 新任务: 手动计划

“资源使用状况”视图

