**<基于PyTorch的前沿深度学习算法集成应用程序接口>**

进度控制与分析报告

**作 者： TeamA全体组员**

**完成日期： 2020.06.10**

**注：全体组员包括张崇智、秦浩桐、黄涵、王茵迪、赵永驰、吴振赫、高明骏**

修订历史记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **修改人员** | **审核人员** |
| 2020.06.10 | V1.0 | 无 | 高明骏 | 张崇智 |
| 2020.06.11 | V1.1 | 无 | 高明骏 | 张崇智 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[一、 实验目标 3](#_Toc42702657)

[二、 实验计划 3](#_Toc42702658)

[2.1 工具 3](#_Toc42702659)

[2.2 具体规划 4](#_Toc42702660)

[三、 实验产出 5](#_Toc42702661)

[四、 总结 10](#_Toc42702662)

# 实验目标

软件项目管理是按需求确定范围、按目标制定项目计划、按计划执行管理的过程，能够增强对软件开发的控制能力，提升软件开发的质量。在本课程中，是对整个软件开发实验各个阶段进行任务管理，主要对需求分析、需求评审、开发与测试、测试评审、配置管理和工作量统计中的任务进行分配。

在项目最初围绕项目编制计划书，并根据计划书编制各阶段计划并实施，并在每阶段结束时检查计划的实际执行情况，分析进度偏差和原因，在此基础上，不断调整、修改计划，直至实验结束。

# 实验计划

## 2.1 工具

OmniPlan是OmniGroup公司旗下的一款专业级的项目管理软件，目前提供了macOS和iOS两种版本，可以用于Mac电脑、iPhone和iPad设备。而其数据同步可以使用官方免费提供的Omni Sync Server，或者可以使用Web DAV，不必担心项目文件不同及时同步的问题。OmniPlan侧重于项目规划，最小单位是成果，成果有的需要几周时间，或几天时间。从整体项目来说，有的小项目一个月的周期，也有的大项目需要几年。相比于Microsoft Project区别主要有以下几点：

1. Microsoft Project 采用左任务列表＋右甘特图的展示模式，非常直观地将与任务相关的所有信息展示出来。OmniPlan也采用了这样的界面，同时提供了资源列表、日历展示等多种视图。从设计语言的角度上看，两者都继承了各自其他产品的风格。
2. OmniPlan为了吸引用户，尽了最大的努力兼容 Microsoft Project，它可以直接打开.mpp文件，也支持导出多种格式为文件，包括.mpp文件。
3. OmniPlan支持便捷生成列表化的数据报告，甚至将Gantt Chart也一并带出，并支持转为.html 格式，非常直观方便。对于多个项目的同时管理，OmniPlan 还推出仪表盘（Dashboard）功能，可以快速查看项目进展情况。

## 2.2 具体规划

首先每周进行两次项目会议，在腾讯会议上进行。第一次是在前一周的周六或周日晚，时长为一到两个小时，主要针对周五老师对当周工作提出的问题和评价进行修改和总结，其次对接下来一周的目标进行明确，具体任务进行调研和分配，任务具体到个人，并对任务的截止时间进行粗略的规定；第二次会议是在当周的周三或周四，依据实验进度具体安排，主要内容是对本周进行的工作进行汇总，并督促未完成的工作，并安排成员对汇总后的结果进行审核，并分配当周汇报任务。

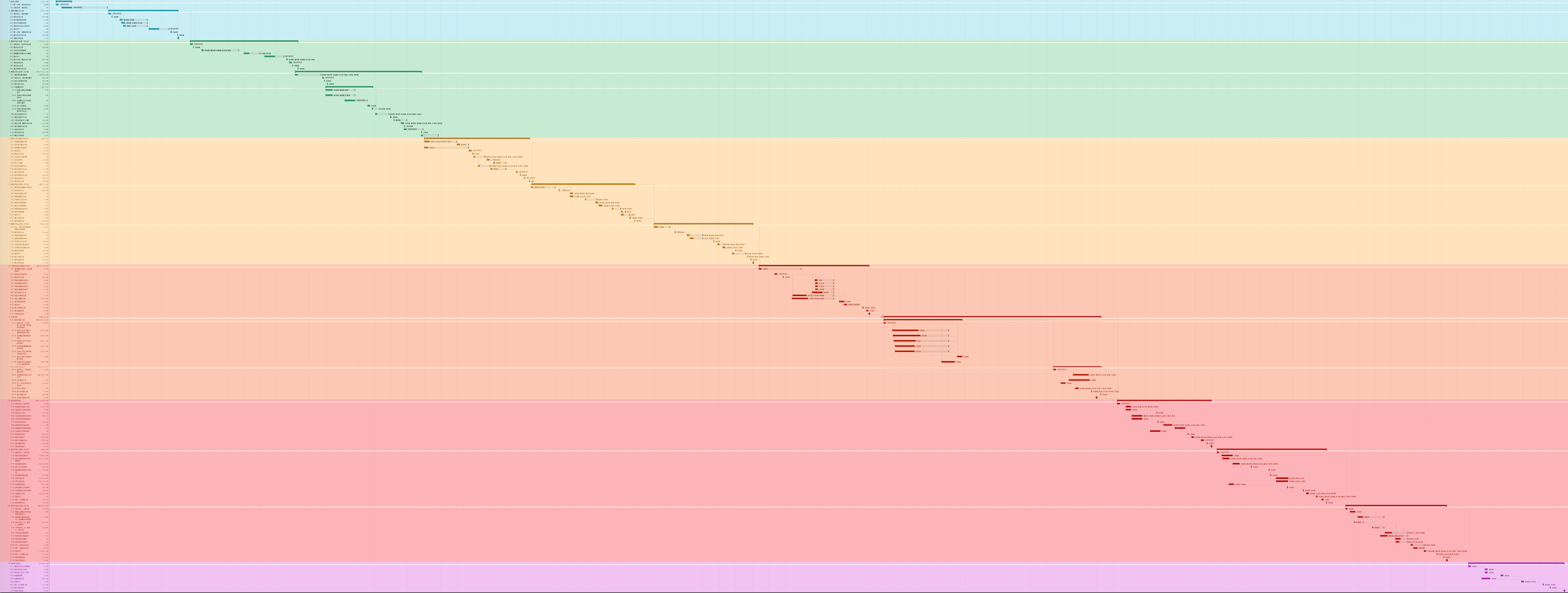
我们使用微信进行平时交流的主要工具，主要进行一些任务更细节上的交流或者是一些成员进度推迟或提前的报备，或是开发和测试过程中一些问题的交流。

使用omniplan进行任务分解后的规范呈现，对于任务截止时间和前后依赖关系进行清晰的说明。同时设定项目计划的基线和里程碑，在到达里程碑后，能够对当前阶段成果进行检查和确认。每周导出截至下周的进度规划，和组员共享并进行沟通交流，调整各个成员的任务计划。

在中期和实验结束前，对当前进度使用Excel绘制燃尽图，对项目计划进行历时评估，同时调整计划。同时，统计各个成员的工时，并与工作量统计的数据进行对比，观察是否有不统一的项，并分析原因。

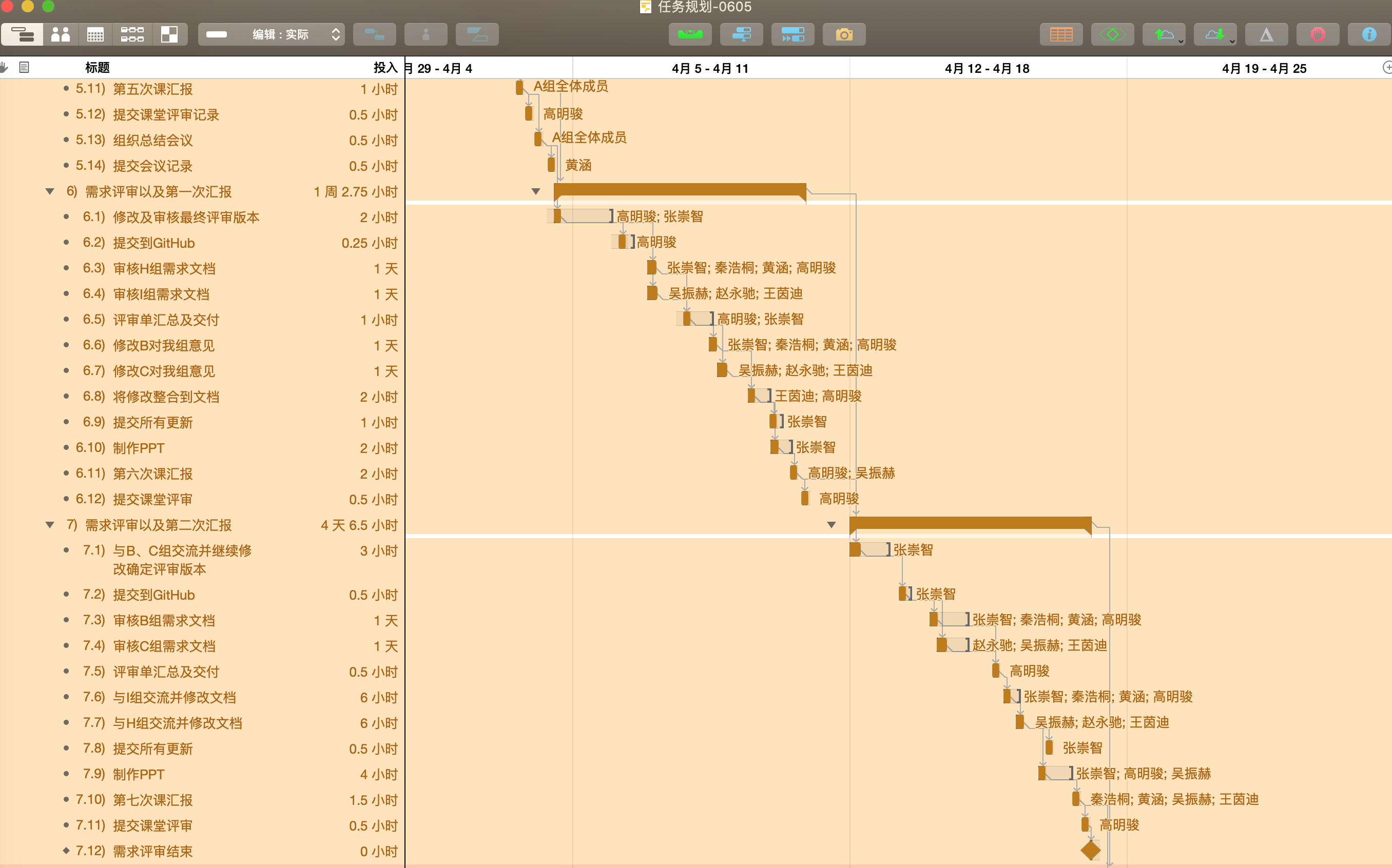
# 实验产出

本节主要介绍，在进度管理过程使用OmniPlan进行控制的产出，主要包括项目甘特图、各成员总结报告以及燃尽图。图1给出了截至6月5日项目的进展整体甘特图。图中所有任务基本具体到个人，以小时为单位，细化到每一个具体产出，如产出会议记录、产出部分需求说明书。以颜色区分不同实验阶段，如青色为需求设计阶段、橙色为需求评审阶段、白色背景为正在进行阶段，每个阶段以里程碑划分。



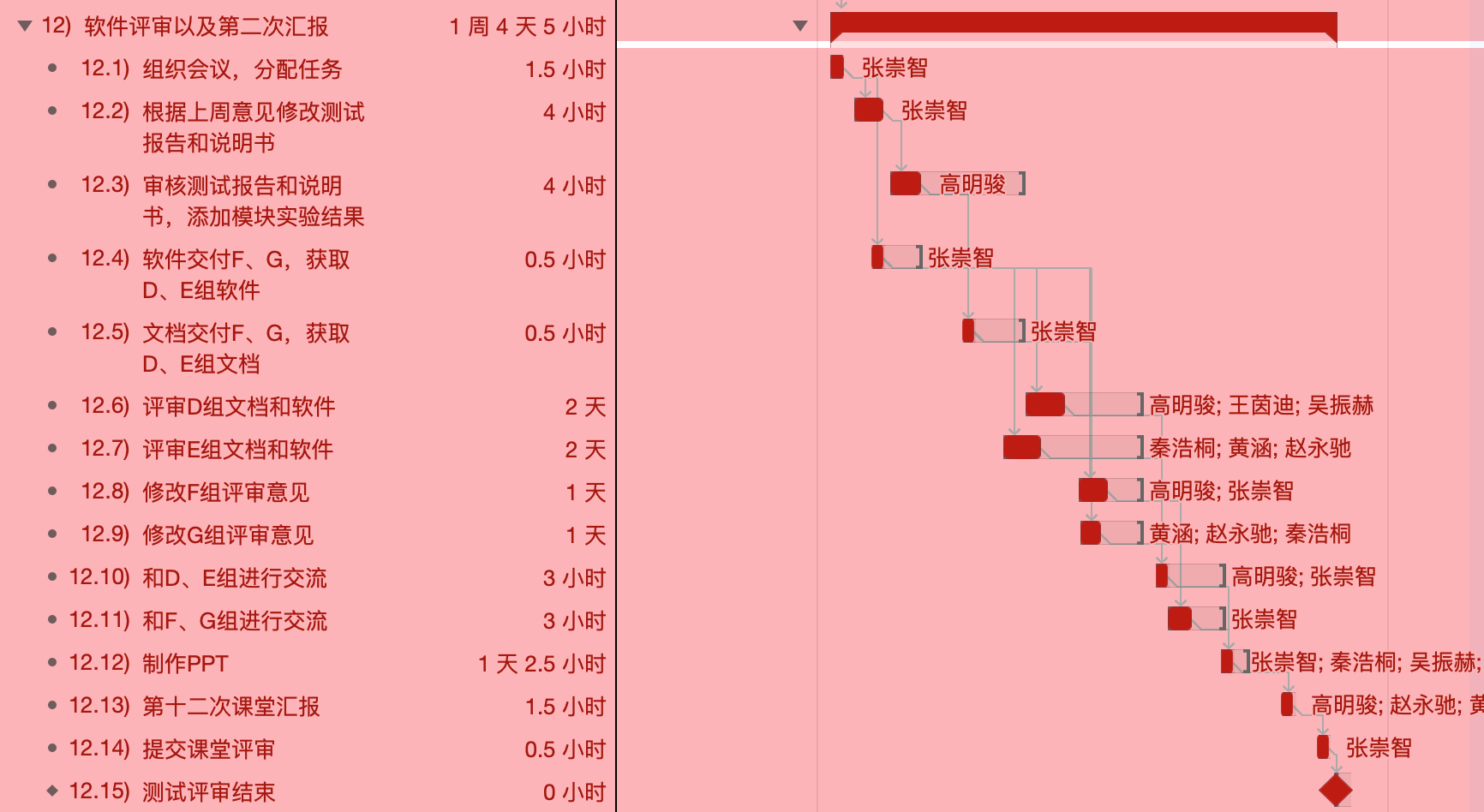
**图1 项目整体甘特图**

图2给出了需求评审阶段的局部甘特图。在需求评审阶段，分成两组并行进行评审修改，每一个任务会记录发布时间，提交时间，成员工作量。由于需求评审阶段是第一次评审，所以评审的组与组之间，评审和被评审之间没有很好的进行任务的并发控制，串行的任务较多，造成了时间上的不充足。



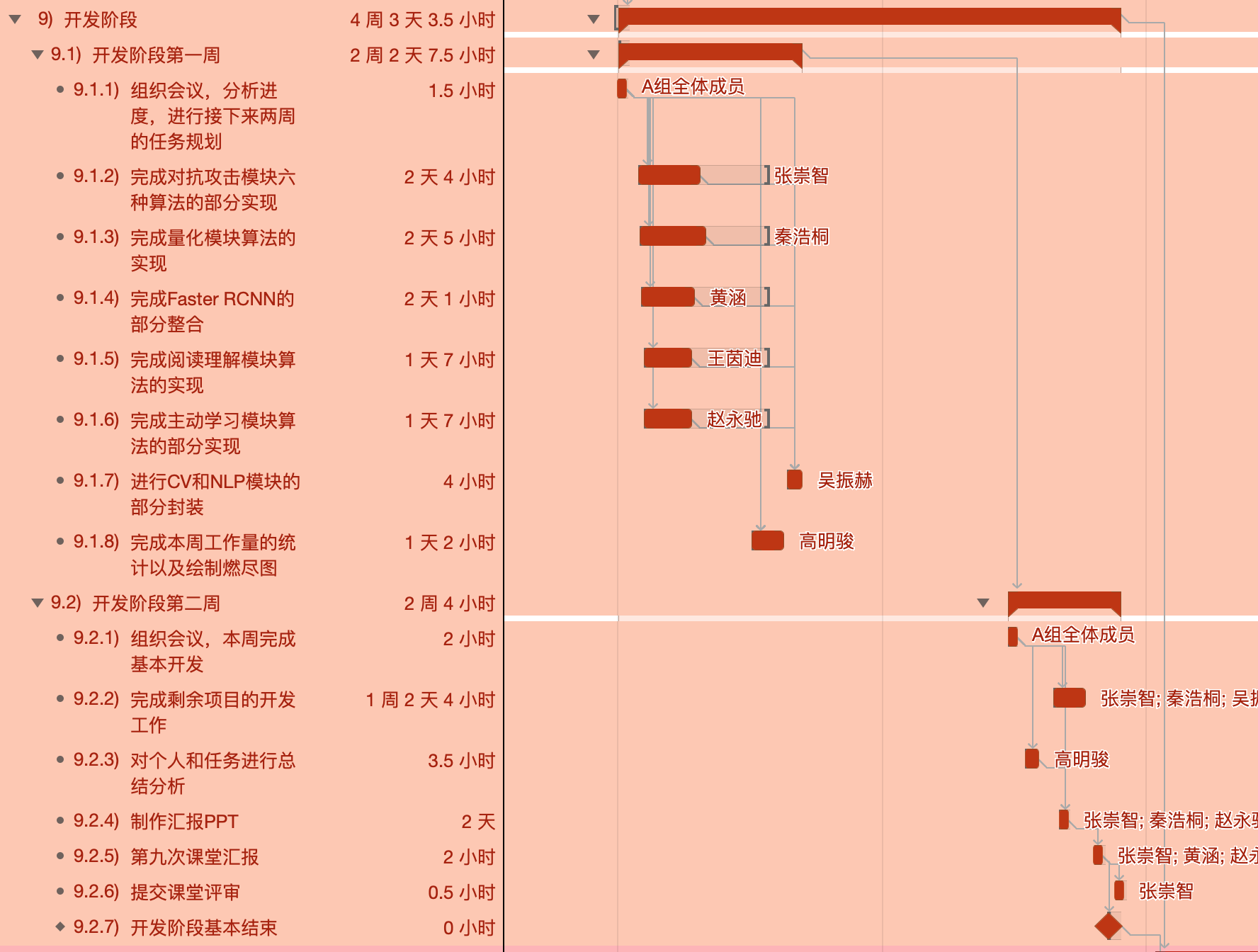
**图2 需求评审阶段局部甘特图**

第14周根据测试评审第一阶段的规划，进行了更具体、合理的分配。测试报告和说明书的更新、与四个组软件和文档的交换、分别进行评审和修改、和其他组进行进行交流，每个步骤前后的依赖关系和并行关系进行了清晰的划分，任务并具体到个人。所有任务的时间节点均以老师的要求为基准线，如对被评审组的意见截止值在周四中午12点。局部甘特图如图3所示。



**图3 测试评审局部甘特图**

图4给出了开发阶段的局部甘特图。开发阶段为第10和11周，CV和NLP两个组分别负责相应模块，模块开发具体到个人。在周末之前初步开发情况汇总至吴振赫，开始集成工作。第11周前半周为五一假期，没有安排工作。后半周在假期结束后进行总结和查漏补缺，要求在周五前完成集成工作，以及部分模块的完善。



**图4 开发阶段局部甘特图**

图5给出了通过Omniplan生成的项目工作报告中的概览，可以看出截至6月6日，整个任务已经进行23周（一天工时为8小时，一周为5天），总计时长914.75h，符合整体项目九百时长左右的预估。具体成员的工作分配和时长统计在工作量分析报告中展示。

****

**图5 项目工作报告概览图**

同时图5给出了每位成员所有任务的总计时长，在图6给出了项目工作报告中的成员任务具体报告的一部分界面，对每位成员具体的工作进行了汇总，其中工时和工作量报告中的成员时长统计并不完全相同，主要原因是工作量报告中并没有将组员参加会议和开发过程时长的统计在内，具体原因在工作量报告说明。图7给出了项目工作报告中的任务报告，对每个任务的具体信息以及和其他任务的相关度进行汇总。

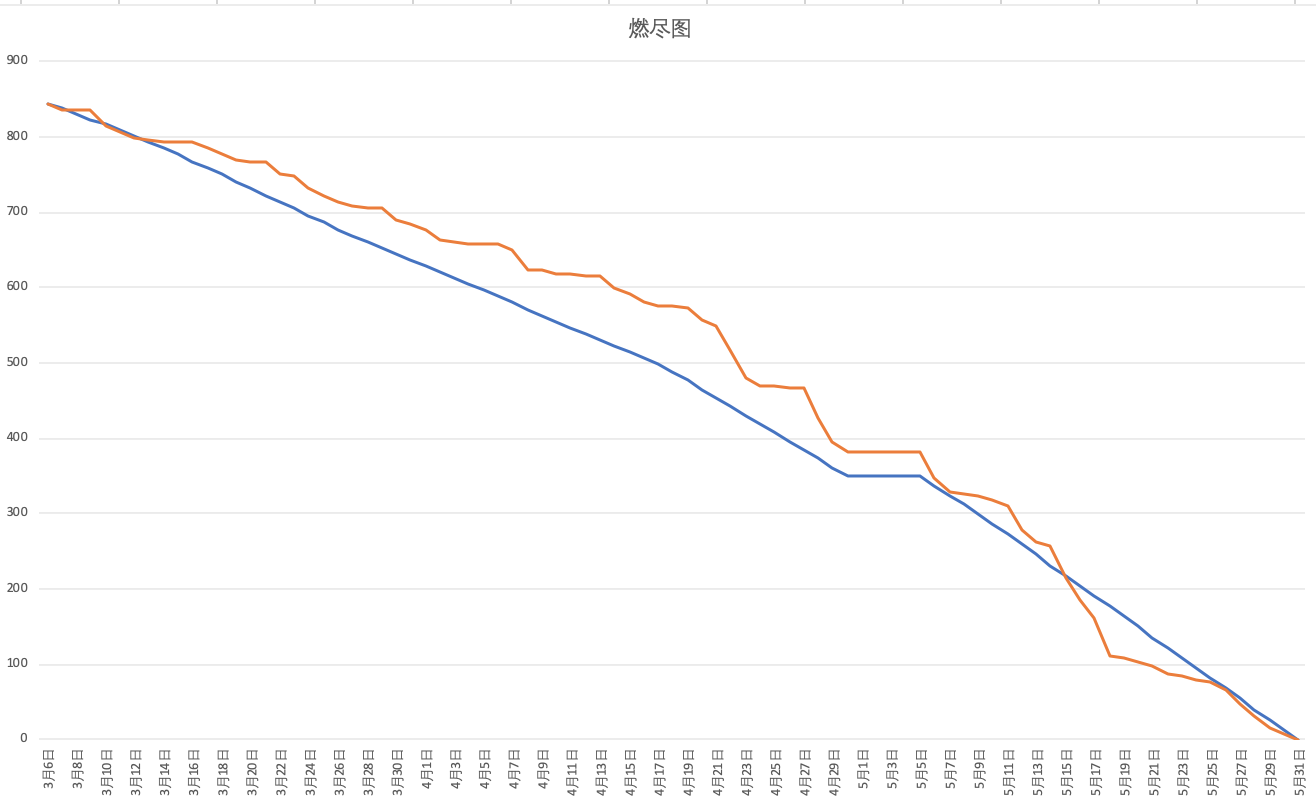
****

**图6 项目工作报告资源报告图**

****

**图7 项目工作报告任务报告图**

图8给出了截至5月30日软件测试阶段结束的项目燃尽图，图中横轴代表日期，由于进度属于每周规划，根据之前的时长统计，我们预计共有投入八百左右时长。图中蓝色折线为预期进度，折线斜率根据每一阶段预计贡献时长不同而不同，橙色线为我们根据任务规划所绘制的燃尽图。从图中可以看出在项目前期，我们始终落后项目预期进度，主要原因是预期中每一阶段包含项目开发的部分，但在实际的进行过程中，由于小组进入状态较慢，主要工时用在完成当前实验中，开发进展缓慢。在五一前后的两周内，我们花费大量时间对产品进一步完善和集成测试等。开发过程中已包含单元测试，在测试阶段的进度只进行功能测试，所以进度小幅度提前。

****

**图8 项目燃尽图**

# 总结

在项目初期需求分析阶段，由于小组成员互相不了解，具体的擅长技能和工作节奏不清晰，在会议中不能畅所欲言各尽所长。之后的任务分配上是组长划分好任务后，大家各自挑选，先到先得。例如小组成员王茵迪上学期修过高等软件工程，由她负责RUCM的绘制，其他成员学习RUCM并分析需求。

需求评审阶段，大家对于整个需求分析的要点还不是很明确，但已经根据研究方向划分了两个小组，分别负责相应模块的开发。两个小组分别对另外两个组进行评审和修改。但由于团队和其他组之间缺乏沟通等因素，导致评审组与组之间的交流十分拖延，而后我们又两个小组分别设置一位小组长来负责总结和汇报主要的进度和意见。

随着项目的进行，实验的分工也越来越明确和熟练，在开发阶段大家都能高质量的完成自己负责部分的编码和测试。对于测试和评审阶段，完成测试报告和测试说明书等各种文档的过程相比于需求阶段的质量更高，被评审出的错误更少，与他组进行交流更熟练。

进度规划、配置管理和工作量统计部分的工作高明骏和张崇智负责，这三个实验需要每周都完成且比较琐碎，在实验过程中，大家也不断摸索更好的规划方式、版本控制方法以及工作量的统计形式。

在整个项目执行过程中，项目的计划和分工很重要，一定要提前明确，同时兼顾工作量和难度，不要出现任务都堆积在一个人身上或者是没有人去完成任务。还要同时定期互相提醒各种任务时间节点，不要出现滞后和拖延，避免多等一的现象发生。在每周任务开始之前，还需要提前去了解一些实验相关知识，避免出现会议上大家现学或是不了解无法发言这种情况。同时一定要保持交流，不要所有问题都在会议上解决，这样也可能造成任务的延后。

总体来说，截至目前，本组在实验过程中存在一些小问题，但都通过不断摸索解决了，也完善了课上老师提出的一些修改意见。最后感谢团队所有成员的付出与合作，感谢老师和同学的修改意见。