

软 件 工 程 实 验

**小众点评**

**编写人：20TeamD**

**编制时间：2020年3月26日**

变更记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 编制人 | 修改说明 | 审核人 |
| 1.0 | 2020.3.26 | 刘思远 | 撰写文档总体框架 |  |
| 1.0 | 2020.3.26 | 许栋亮、黄雨鹤 | 撰写项目总体计划 |  |
| 1.1 |  |  |  |  |

目录

[一 进度控制目的 4](#_Toc36127312)

[二 进度控制工具 4](#_Toc36127313)

[三 软件项目进度控制步骤 4](#_Toc36127314)

[3.1 总体计划 4](#_Toc36127315)

[3.2 计划实际完成情况及变化 7](#_Toc36127316)

[1) 实验准备(2020.3.8-2020.3.18) 7](#_Toc36127317)

[2) 实验一:项目需求分析(2020.3.19-2020.3-26) 8](#_Toc36127318)

[四 资源分配及其影响因素分析 8](#_Toc36127319)

[4.1 资源分配原则 8](#_Toc36127320)

# 一 进度控制目的

为了保障项目能按期完成，在项目完成过程中能够合理安排资源供应，同时能够节约项目成本。进度控制是在保障项目质量，不增加实际成本的前提下，最大限度的确保项目工期能够满足预计项目工期的要求，这是进度控制的主要目的。

# 二 进度控制工具

Teambition是一个国内自主开发的项目管理工具软件，是国内团队协作工具的创导者。其开发目的是让团队协作更加简单高效，帮助项目管理者实现时间、资源、成本的计划、控制。

Teambition是国内团队协作工具的创导者，通过帮助团队轻松共享和讨论工作中的任务、文件、分享、日程等内容，让团队协作焕发无限可能。Teambition在网页、桌面、移动环境都打造了体验出众的应用，所以用户随时随地都可以和团队协作，其iPhone应用还被苹果公司评为2015年度最佳应用。目前，已经有超过一百万用户通过Teambition进行团队协作，其中也包括多个行业的龙头企业。Teambition公司已获得来自戈壁投资、IDG、盘古及北极光的多轮融资，并于2014年入选中国最具投资价值企业50强，2019年3月26日Teambition被阿里巴巴全资收购。

# 三 软件项目进度控制步骤

# 3.1 总体计划

在确定课程项目以及相关实验内容之后，小组进行开会讨论，对实验内容进行分析，并依据项目属性，基于Teambition工具对整个实验过程做了整体规划，如表3-1所示：

|  |  |  |  |
| --- | --- | --- | --- |
| 任务名称 | 计划工期 | 计划开始时间 | 计划完成时间 |
| **项目确定以及介绍** | **3 个工作日** | **2020年3月8日** | **2020年3月11日** |
| 项目调研 | 8 工时 | 2020年3月8日 | 2020年3月8日 |
| 选题会议 | 21 工时 | 2020年3月10日 | 2020年3月10日 |
| PPT制作 | 3 工时 | 2020年3月11日 | 2020年3月11日 |
| **项目计划书** | **3 个工作日** | **2020年3月16日** | **2020年3月18日** |
| 项目计划会议 | 21 工时 | 2020年3月16日 | 2020年3月16日 |
| 项目简介 | 3 工时 | 2020年3月16日 | 2020年3月16日 |
| 项目概述 | 3 工时 | 2020年3月16日 | 2020年3月16日 |
| 项目团队组织 | 3 工时 | 2020年3月17日 | 2020年3月17日 |
| 实施计划 | 3 工时 | 2020年3月17日 | 2020年3月17日 |
| 计划书修改与整合 | 14 工时 | 2020年3月17日 | 2020年3月17日 |
| Teambition学习与使用 | 14 工时 | 2020年3月18日 | 2020年3月18日 |
| **软件需求分析** | **7 个工作日** | **2020年3月19日** | **2020年3月26日** |
| 需求分析会议 | 21 工时 | 2020年3月19日 | 2020年3月19日 |
| 软件模块分析 | 4 个工作日 | 2020年3月19日 | 2020年3月21日 |
| 测试需求分析 | 4 个工作日 | 2020年3月22日 | 2020年3月23日 |
| 软工规范分析 | 2 个工作日 | 2020年3月24日 | 2020年3月24日 |
| 扩展功能分析 | 4 个工作日 | 2020年3月25日 | 2020年3月26日 |
| 需求分析会议 | 21 工时 | 2020年3月26日 | 2020年3月26日 |
| **软件需求评审** | **9 个工作日** | **2020年3月28日** | **2020年4月5日** |
| 需求评审会议 | 21 工时 | 2020年3月28日 | 2020年3月28日 |
| 修改软件模块需求 | 4 个工作日 | 2020年3月28日 | 2020年3月29日 |
| 修改测试需求 | 4 个工作日 | 2020年3月30日 | 2020年3月31日 |
| 修改软工规范需求 | 4 个工作日 | 2020年4月1日 | 2020年4月2日 |
| 修改扩展功能需求 | 4 个工作日 | 2020年4月3日 | 2020年4月4日 |
| 需求评审会议 | 21 工时 | 2020年4月5日 | 2020年4月5日 |
| **软件需求复评审** | **4 个工作日** | **2020年4月6日** | **2020年4月9日** |
| 需求复审会议 | 21 工时 | 2020年4月6日 | 2020年4月6日 |
| 确定软件模块需求 | 2 个工作日 | 2020年4月6日 | 2020年4月6日 |
| 确定测试需求 | 2 个工作日 | 2020年4月7日 | 2020年4月7日 |
| 确定软工规范需求 | 2 个工作日 | 2020年4月8日 | 2020年4月8日 |
| 确定扩展功能需求 | 2 个工作日 | 2020年4月9日 | 2020年4月9日 |
| 需求复审会议 | 21 工时 | 2020年4月9日 | 2020年4月9日 |
| **项目进度展示(一)** | **30个工作日** | **2020年3月8日** | **2020年4月16日** |
| 工作日志 | 30 个工作日 | 2020年 3月8日 | 2020年 4月16日 |
| 进度控制分析报告 | 30 个工作日 | 2020年 3月8日 | 2020年 4月16日 |
| 项目部署 | 30 个工作日 | 2020年 3月8日 | 2020年 4月16日 |
| 实验分析报告 | 30 个工作日 | 2020年 3月8日 | 2020年 4月16日 |
| 工作量估计 | 30 个工作日 | 2020年 3月8日 | 2020年 4月16日 |
| **软件设计与实现** | **17 个工作日** | **2020年4月18日** | **2020年5月7日** |
| 软**件**设计会议 | 21 工时 | 2020年4月18日 | 2020年4月18日 |
| 软件设计 | 9 个工作日 | 2020年4月18日 | 2020年4月20日 |
| 实现分工会议 | 21 工时 | 2020年4月21日 | 2020年4月21日 |
| 软件实现（一） | 21个工作日 | 2020年4月22日 | 2020年4月28日 |
| 问题讨论会议 | 21 工时 | 2020年4月29日 | 2020年4月29日 |
| 软件实现（二） | 21个工作日 | 2020年4月30日 | 2020年5月6日 |
| 软件展示 | 21 工时 | 2020年5月7日 | 2020年5月7日 |
| **软件测试分析** | **6 个工作日** | **2020年5月9日** | **2020年5月14日** |
| 软件测试会议 | 21 工时 | 2020年5月9日 | 2020年5月9日 |
| 测试需求分析 | 10 个工作日 | 2020年5月10日 | 2020年5月14日 |
| **软件测试评审** | **5个工作日** | **2020年5月16日** | **2020年5月21日** |
| 测试评审会议 | 21 工时 | 2020年5月16日 | 2020年5月16日 |
| 修改测试需求 | 8 个工作日 | 2020年5月17日 | 2020年5月21日 |
| **软件测试复评审** | **5 个工作日** | **2020年5月23日** | **2020年5月28日** |
| 测试复评审会议 | 21 工时 | 2020年5月23日 | 2020年5月23日 |
| 修改测试需求 | 10 个工作日 | 2020年5月24日 | 2020年5月28日 |
| **软件演示与测评** | **5 个工作日** | **2020年5月30日** | **2020年6月4日** |
| 软件演示会议 | 21 工时 | 2020年5月30日 | 2020年5月30日 |
| 软件演示与测评 | 10个工作日 | 2020年5月31日 | 2020年6月4日 |
| **项目进度展示(二)** | **38 个工作日** | **2020年4月17日** | **2020年6月11日** |
| 工作日志 | 38 个工作日 | 2020年 4月17日 | 2020年 6月11日 |
| 进度控制分析报告 | 38个工作日 | 2020年 4月17日 | 2020年 6月11日 |
| 项目部署 | 38 个工作日 | 2020年 4月17日 | 2020年 6月11日 |
| 实验分析报告 | 38 个工作日 | 2020年 4月17日 | 2020年 6月11日 |
| 工作量估计 | 38 个工作日 | 2020年 4月17日 | 2020年 6月11日 |
| **实验分析与总结** | **6个工作日** | **2020年6月13日** | **2020年6月18日** |
| 会议 | 21 工时 | 2020年6月13日 | 2020年6月13日 |
| 综合实验分析 | 4 个工作日 | 2020年6月13日 | 2020年6月18日 |
| 综合实验总结 | 4 个工作日 | 2020年6月13日 | 2020年6月18日 |
| **每周组会** | **12 个工作日** | **2020年3月16日** | **2020年6月18日** |

表3-1 项目总体计划

从项目总体计划可以看出，项目计划工期总共128个工作日。在工时内我们通过对Teambition上的任务完成情况以及各类变化因素的影响，我们可以得到完整的计划完成情况与实验发生变化的原因及变化的实际完成结果。

# 3.2 计划实际完成情况及变化

根据对实验过程中的历次Teambition的任务完成情况，我们得出了具体的计划完成情况及变化因素。具体情况如下所示。

1. 实验准备(2020.3.8-2020.3.18)

实验准备在Teambition上无具体任务项。主要为项目的确定与调研以及项目计划书的制作。实验准备最后按时完成，组员们协调完成了项目的调研与计划书的编写。具体情况如图3-1所示，该图截取自的变更记录部分。



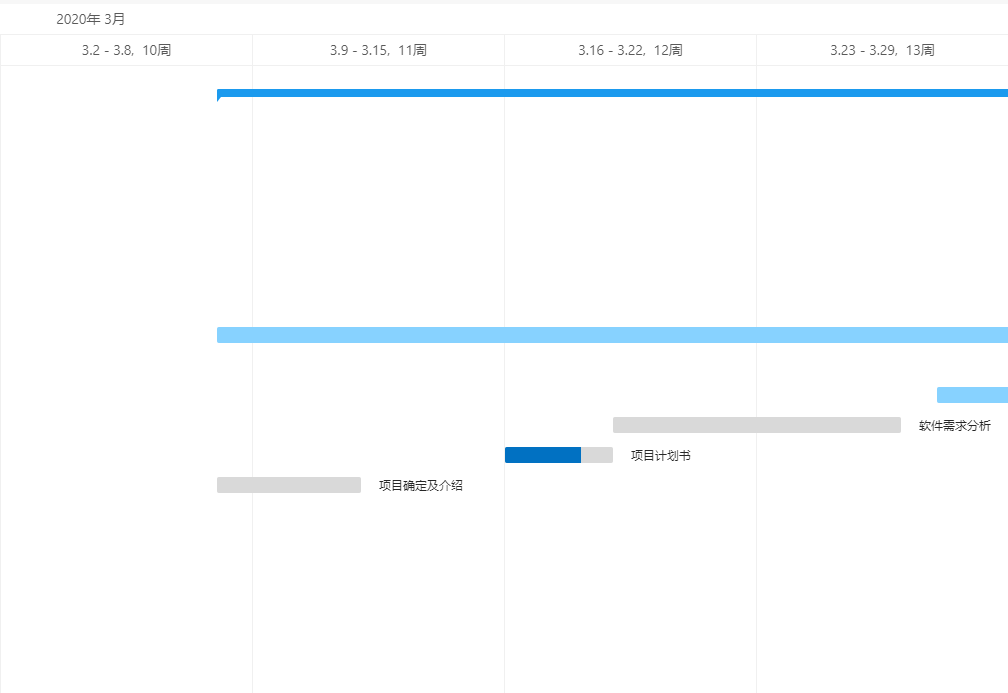
图3-1 计划书完成情况

1. 实验一:项目需求分析(2020.3.19-2020.3-26)

项目的需求分析任务在teambition上有具体的任务，并且被分为五个子任务。截止3.26日项目需求分析的各项任务均已完成，需求分析文档也已经撰写完成。具体内容如图3-2所示.



图3-2 Teambition上软件需求分析具体完成情况



图

# 四 资源分配及其影响因素分析

# 4.1 资源分配原则

为了更好更快的完成实验内容，小组采用的是按照个人特点分配任务，在项目确定之后，小组组内进行多次会议，在会议中每个人阐述自己的优势部分，根据个人不同的优势，具体任务分配如表4-1所示。

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | | 性别 | | 特点 | | 分工 | | 职位 | |
| 许栋亮 | | 男 | | 1有C++开发经验。 | | 1.负责小组需求分析文档的指导与撰写  2.负责项目总体计划的安排 | | 组长 | |
| 刘新宇 | | 男 | | 1.有工程开发经验  2.熟悉C++、python | | 1.负责小组软件开发。  2.负责实验8 | | 组员 | |
| 黄雨鹤 | | 男 | | 1.有对EOS平台上的程序的测试经验。 | | 1.负责小组测试文档的撰写与指导工作。  2.负责项目总体计划的安排 | | 组员 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 林讯 | 男 | 1.有Spring Boot、Vue-cli开发经验. | 1.负责前端交互界面的开发 | 组员 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 叶明林 | 男 | 1.熟悉JAVA、Python。  2.了解C++。  3.有Spring Boot、Vue.js和Android开发经验。 | 1.负责项目小组软件开发.  2.负责实验7的撰写与后续跟进。 | 组员 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 陈瀚清 | 男 | 1.有C++开发经验。 | 1.负责项目小组的需求分析文档撰写。 | 组员 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 刘思远 | 男 | 1.有C++开发经验。 | 1.负责项目软件测试。  2.负责实验6文档的跟进与分析。 | 组员 |

表 4-1分配原则

对于实验的整个过程而言，每个组员对于每次实验都有参与，表4-1的分工只是某位成员在某次实验可能更适合某次实验，并且在该实验中负责某一重要模块，且担负对其他组员工作的指导作用。