**软件需求规格说明书**

基于MPXJ的\*.mpp文件信息读写的JAVA桌面应用——Mini Project

小组成员：

SY1506104 刘克瑞

SY1506106 詹鹏飞

SY1506114 陈阳

SY1506115 彭柯宾

文件修改记录

| 修改日期 | 版本 | 修改页码、章节、条款 | 修改描述 | 作者 |
| --- | --- | --- | --- | --- |
| 2016.4.4 | 2.0 | 需求说明书副标题  2.2用户类型分析  3.2用例说明 | 根据课上老师提出的问题进行了修改，主要包括确定并添加软件名称，重新界定了用户类型。 | 陈阳 |
| 2016.4.8 | 3.0 | 需求说明书标题,  1.2需求分析理论,  2.1产品信息描述  3.1用例图标题  3.2用例说明101  3.2用例说明自动计划及手动计划 | 主要根据课上老师提出的问题进行了修改  在用例说明101中插入若未能打开.mpp文件的解决方法  在用例说明后插入自动计划及手动计划的注释。 | 詹鹏飞 |
| 2016.4.12 | 4.0 | 2.2用户类型分析 | 删除了用户类型分析一节 | 刘克瑞 |
| 2016.5.03 | 5.0 | 3.2 用例说明  6 问题及解决方案 | 增加的每个用例的用例场景，使得需求具体化。  对存需求中存在的问题和相应的解决方案进行的描述 | 刘克瑞 |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1引言 4](#_Toc446516691)

[1.1编写目的 4](#_Toc446516692)

[1.2软件需求分析理论 4](#_Toc446516693)

[1.3项目范围 4](#_Toc446516694)

[1.4参考资料 4](#_Toc446516695)

[2总体概述 5](#_Toc446516696)

[2.1产品描述 5](#_Toc446516697)

[2.2用户类型分析 5](#_Toc446516698)

[2.3运行环境 5](#_Toc446516699)

[2.3.1目标及架构 5](#_Toc446516700)

[2.3.2 主机 5](#_Toc446516701)

[2.4设计和实现的约束条件 6](#_Toc446516702)

[2.5假设和依赖 6](#_Toc446516703)

[3功能需求 7](#_Toc446516704)

[3.1用例图 7](#_Toc446516705)

[3.2用例说明 8](#_Toc446516706)

[4外部接口需求 20](#_Toc446516707)

[4.1硬件接口 20](#_Toc446516708)

[4.2软件接口 20](#_Toc446516709)

[4.3用户界面 20](#_Toc446516710)

[5非功能性需求 20](#_Toc446516711)

[5.1性能需求 20](#_Toc446516712)

[5.2安全性需求 20](#_Toc446516713)

# 1引言

## 1.1编写目的

为明确软件需求、安排项目规划与进度、组织软件开发与测试，撰写本文档。

## 1.2软件需求分析理论

软件需求分析（Software Reguirement Analysis）是对用户想要实现的功能进行分析，理解用户想要得到的该软件的完整功能，为将来进行软件设计提供一个基本依据。

需求分析的具体内容可以归纳为六个方面：软件的功能需求，软件与硬件或其他外部系统接口，软件的非功能性需求，软件的反向需求，软件设计和实现上的限制，阅读支持信息。

软件需求分析的主要实现目标：

1) 对实现软件的功能做全面的描述，帮助用户判断实现功能的正确性、一致性和完整性，促使用户在软件设计启动之前周密地、全面地思考软件需求；

2) 了解和描述软件实现所需的全部信息，为软件设计、确认和验证提供一个基准；

3) 为软件管理人员进行软件成本计价和编制软件开发计划书提供依据；

## 1.3项目范围

项目名称：基于MPXJ的mpp文件信息读取的JAVA桌面应用Mini Project

分析系统名称：MPXJ

## 1.4参考资料

1. 《软件工程基础》  赵一丁 北京邮电大学出版社

2. 《软件需求》  劳森 (作者), 刘晓晖 (译者) 电子工业出版社

3. 《软件需求工程：原理和方法》  金芝，刘璘，金英  科学出版社

4. 《实用软件工程》第三版  殷人昆 清华大学出版社

# 2总体概述

## 2.1产品描述

Microsoft Project（或MSP）是由微软开发销售的[项目管理软件](http://baike.baidu.com/view/114717.htm" \t "_blank)程序。软件设计目的在于协助项目经理发展计划、为任务分配资源、跟踪进度、管理预算和分析工作量。我们预计设计的Mini Project是Microsoft Project的一个轻量级实现，在保证Microsoft Project基本功能的基础上，根据用户需求新增部分功能实现，如下

1提醒功能，设置提醒后系统会在指定时间提醒用户

2根据用户图表拖拽自动改变计划表的相应的值，提升用户体验

3将任务量，任务时间等图形化，用图表表示出来，提升用户体验，利于统计任务数据

## 2.2运行环境

### 2.2.1目标及架构

ARM：ARM9, ARM11, ARM11 MPCore, ARM Cortex A8, ARM Cortex A9

ColdFire：ColdFire v2, ColdFire v3, ColdFire v4e   
Intel：Pentium family (Pentium, Pentium Pro, Pentium II, Pentium III, Pentium 4,

Pentium M, Xeon, Xeon LV, Core, Core 2 Duo, Atom

### 2.2.2 主机

Windows Vista （商用和企业版 ) SP2   
Windows 7   
Red Hat Enterprise Linux Workstation 4, Update 6 或 8, x86 (32-bit)   
Red Hat Enterprise Linux Workstation 5, Update 2 或 3, x86 (32-bit/64-bit)   
Red Hat Fedora 9, x86-64   
Red Hat Fedora 11, x86(32-bit/64-bit)

## 2.3设计和实现的约束条件

**CO-1**:系统设计采用Java语言进行开发，使用MPXJ类包来创建、读写Microsoft Project Exchange (MPX)文件，Project 2002 和 2003 MSPDI XML 文件, 与读起取Microsoft Project 98, 2000, 2002, 和 2003 (MPP) 文件

**CO-2**：系统开发进度需严格按照本次软件工程综合实验标准

## 2.4假设和依赖

暂无

# 3功能需求

## 3.1用例模型

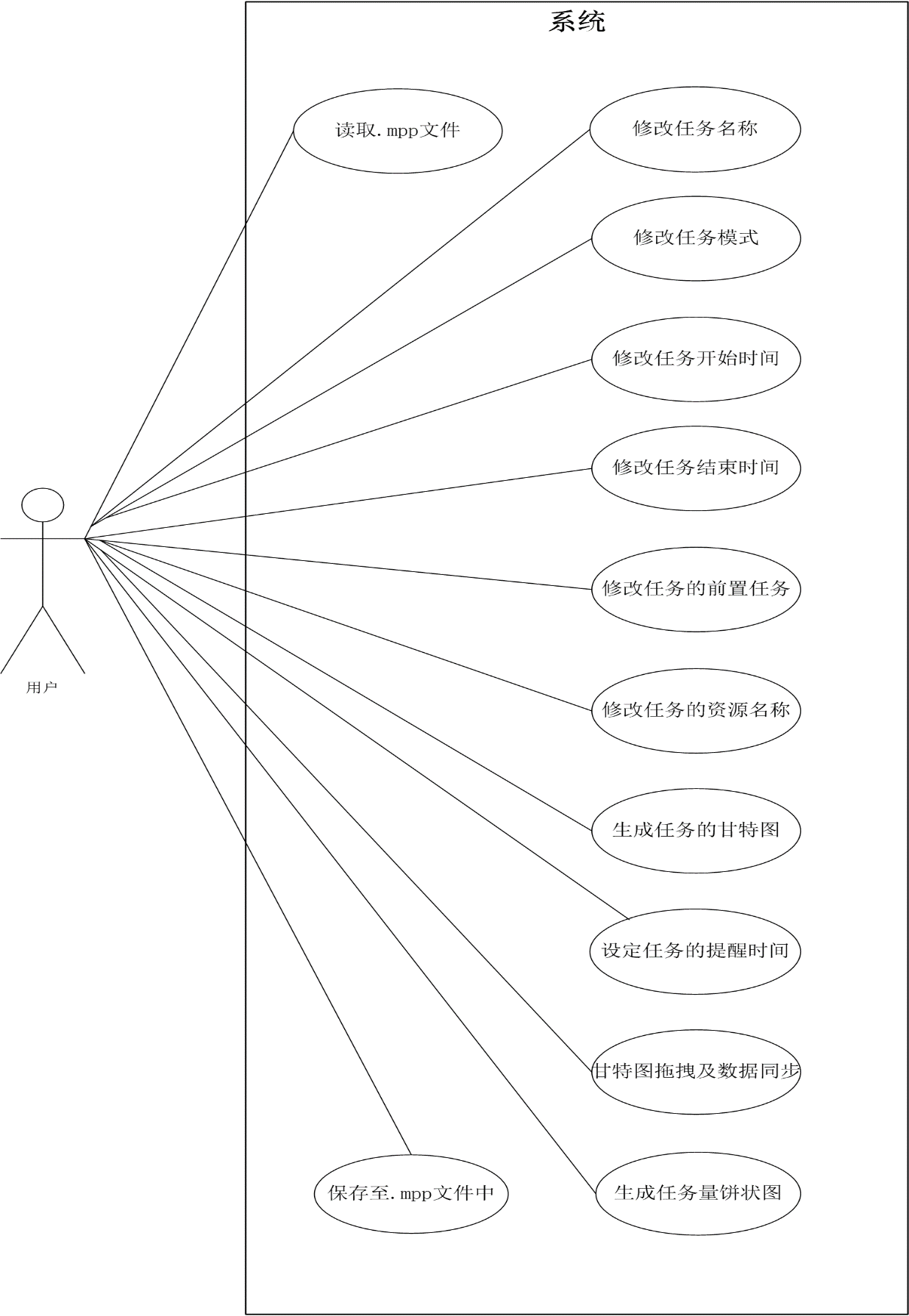


图1 用例图

## 3.2用例说明

**用例101 读取.mpp文件**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 读取.mpp文件 | |
| Use Case ID | 101 | |
| Brief Description | 用户读取计划表 | |
| Precondition | 系统正常打开运行，无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Steps1 | 用户使用Mini Project打开.mpp文件 |
| Steps2 | 系统显示project计划表 |
| Postcondition | 生成该.mpp文件内容的视图 |
| Specific Alternative Flows | RFS | 1 |
| Steps1 | 如果用户未能成功打开.mpp文件，则弹出报错窗口 |
| Steps2 | 回到Mini Project主页面 |
| Postcondition | 未能打开.mpp文件 |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户在没有安装MicroSoft Project的情况下想要打开mpp文件时，可以使用Mini Project 来作为工具打开和读取mpp文件。

**用例102 修改任务名称**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 修改任务名称 | |
| Use Case ID | 102 | |
| Brief Description | 用户可修改计划表中任何任务名称 | |
| Precondition | 系统读取.mpp文件无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Steps1 | 用户双击想要修改的任务名称所在的单元格 |
| Steps2 | 修改任务名称 |
| Postcondition | 该任务名称被修改 |
| Specific Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件之后，想要修改具体任务的名称时，可以选定想要修改的任务直接进行修改。

**用例103 修改任务模式**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 修改任务模式 | |
| Use Case ID | 103 | |
| Brief Description | 用户可将任务模式修改为自动计划或手动计划 | |
| Precondition | 系统读取.mpp文件无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户双击任务模式 |
| Step2 | 用户选择自动计划或手动计划 |
| Postcondition | 任务模式显示为自动计划或手动计划 |
| Specific Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

注：自动计划：指的是分配完子任务的时候父任务按时间顺序自动分配时间长度。

手动计划：指的是分配子任务以及父任务的时候均手动安排时间节点。

用例场景：当用户使用Mini Project 成功打开mpp文件之后，想要修改任务模式时，可以在任务模式的选项中直接修改为自动计划或手动计划。

**用例104 修改任务开始时间**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 修改任务开始时间 | |
| Use Case ID | 104 | |
| Brief Description | 用户可设定年月日作为任务开始时间 | |
| Precondition | 系统读取.mpp文件无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户双击任务开始时间 |
| Step2 | 用户输入年月日 |
| Postcondition | 任务开始时间显示为用户修改的时间 |
| Specific Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件之后，想要修改任务的开始时间，可以选定相应任务的开始时间直接进行修改。修改内容包括年、月、日。

**用例105 修改任务结束时间**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 修改任务结束时间 | |
| Use Case ID | 105 | |
| Brief Description | 用户可设定年月日作为任务结束时间 | |
| Precondition | 系统读取.mpp文件无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户双击任务结束时间 |
| Step2 | 用户输入年月日 |
| Postcondition | 任务结束时间显示为用户修改的时间 |
| Specific Alternative Flows | RFS | Basic Flow Step2 |
| Step1 | 用户输入年月日早于任务开始时间 |
| Step2 | 系统提示错误 |
| Postcondition | 系统提示错误，修改失败 |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件之后，想要修改任务的结束时间，可以选定相应任务的结束时间直接进行修改。修改内容包括年、月、日。

**用例106 修改任务的前置任务**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 修改任务的前置任务 | |
| Use Case ID | 106 | |
| Brief Description | 用户可设定某一任务为另一任务的前置任务 | |
| Precondition | 系统读取.mpp文件无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户双击某一任务的前置任务所在单元格 |
| Step2 | 用户输入任务标号 |
| Postcondition | 该任务前置任务显示为用户所输入的任务标号 |
| Specific Alternative Flows | RFS | Basic Flow Step2 |
| Step1 | 用户输入任务标号不存在 |
| Step2 | 系统提示错误 |
| Postcondition | 系统提示错误，修改失败 |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件之后，发现某个任务的前置任务有误，想要修改时，可以选定相应任务的“前置任务”一栏直接进行修改。

**用例107 修改任务的资源名称**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 修改任务的资源名称 | |
| Use Case ID | 107 | |
| Brief Description | 用户可设定某些资源为任务的资源名称 | |
| Precondition | 系统读取.mpp文件无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户双击某一任务的资源名称所在单元格 |
| Step2 | 用户输入资源名称 |
| Postcondition | 该任务资源名称显示为用户所输入的资源名称 |
| Specific Alternative Flows | RFS |  |
| Step1 |  |
| Step2 |  |
| Postcondition |  |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件之后，想要修改任务的的资源时，可以选定任务对应的任务资源进行修改。

**用例108 生成任务的甘特图**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 生成任务的甘特图 | |
| Use Case ID | 108 | |
| Brief Description | 根据任务属性生成相应的甘特图 | |
| Precondition | 系统读取.mpp文件无异常，任务各个属性无错误 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户单击“甘特图”按钮 |
| Step2 | 系统生成相应的甘特图，并显示 |
| Postcondition | 系统生成相应的甘特图显示界面 |
| Specific Alternative Flows | RFS | Basic Flow Step2 |
| Step1 | 任务关键属性为空 |
| Step2 | 系统提示错误 |
| Postcondition | 生成失败 |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件之后，想要查阅计划所对应的甘特图时，可以点击“甘特图”按钮，查阅计划相对应的甘特图。

**用例109 设定任务的提醒时间**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 设定任务的提醒时间 | |
| Use Case ID | 109 | |
| Brief Description | 用户可设定具体时间，系统在当天会提醒用户 | |
| Precondition | 系统读取.mpp文件无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户双击某一任务的提醒时间所在单元格 |
| Step2 | 用户输入具体时间，精确到分钟 |
| Postcondition | 该任务提醒时间显示为用户所输入的具体时间 |
| Specific Alternative Flows | RFS | Basic Flow Step2 |
| Step1 | 用户输入时间不在任务开始时间和结束时间之间 |
| Step2 | 系统提示时间错误 |
| Postcondition | 设定提醒时间失败 |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件之后，想要给相应的任务设置提醒功能时，可以再“提醒”一栏设置相应的时间，精确到秒。

**用例110 拖拽甘特图，可同步到任务表中**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 拖拽甘特图，可同步到任务表中 | |
| Use Case ID | 110 | |
| Brief Description | 用户拖拽甘特图，调整任务属性，可同步到任务表中 | |
| Precondition | 系统读取.mpp文件无异常，生成甘特图无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户拖拽或修改甘特图中某一任务的属性 |
| Step2 | 系统同步到相应的任务表中 |
| Postcondition | 甘特图和任务表中任务的属性被同步修改 |
| Specific Alternative Flows | RFS | Basic Flow Step1 |
| Step1 | 甘特图中任务属性修改错误 |
| Step2 | 系统提示错误 |
| Postcondition | 任务表和甘特图修改失败 |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件之后，觉得修改任务的相应信息不方便时，可以直接对计划的甘特图进行拖拽，相应的数据会发生改变。

**用例111 生成任务量直方图图**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 生成任务量直方图 | |
| Use Case ID | 111 | |
| Brief Description | 系统生成任务量直方图，图形化各个资源的任务量 | |
| Precondition | 系统读取.mpp文件无异常，任务资源已分配 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户单击“任务量直方图” |
| Step2 | 系统生成任务量直方图 |
| Postcondition | 生成成功并显示 |
| Specific Alternative Flows | RFS | Basic Flow Step1 |
| Step1 | 资源尚未分配 |
| Step2 | 系统提示错误 |
| Postcondition | 生成失败 |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件之后，想要查阅任务的相应资源的使用情况时，可以点击“任务资源直方图”进行查阅。

**用例112 保存至.mpx文件中**

|  |  |  |
| --- | --- | --- |
| Use Case Name | 保存至.mpx文件中 | |
| Use Case ID | 112 | |
| Brief Description | 用户保存修改的内容 | |
| Precondition | 系统读取.mpp文件无异常，修改无异常 | |
| Primary Actor | 用户 | |
| Secondary Actors | 无 | |
| Dependency | 无 | |
| Generalization | 无 | |
| Basic Flow | Step1 | 用户单击保存按钮 |
| Step2 | 系统保存至相应.mpx文件中 |
| Postcondition | .mpp文件内容被修改并保存 |
| Specific Alternative Flows | RFS |  |
| Step1 |  |
| Step2 |  |
| Postcondition |  |
| Global Alternative Flows | 无 | |
| Steps(numbered) |  |
| Postcondition |  |
| Bounded Alternative Flows | 无 | |
| RFS |  |
| Steps(numbered) |  |
| Postcondition |  |

用例场景：当用户使用Mini Project 成功打开mpp文件，进行各种操作之后想要保存时，点击保存，选择文件格式为mpx，就可以保存修改的内容了。

# 4外部接口需求

## 4.1硬件接口

暂无

## 4.2软件接口

1.操作系统：Windows Vista （商用和企业版 ) SP2 ，Windows 7，Windows 8.1

2.主要工具：Java提供的MPXJ类包

## 4.3用户界面

待定

# 5非功能性需求

## 5.1性能需求

PE-1：系统开始读取文件到显示给用户的响应时间控制在2秒内

PE-2：用户点击生成甘特图后，系统响应时间控制在2秒内

PE-3：用户点击生成任务量饼状图后，系统响应时间控制在2秒内

## 5.2安全性需求

暂无

# 6问题及解决方案

## 6.1 存在的问题

1. 用户需求与软件的功能之间不能保证完全的一致，多少会有一些的差异。

2. 软件的功能是否能正常展现还不能确定。

## 6.2 解决方案

1．根据用户需求不断完善功能的同时，就需求和用户进行沟通和协商，争取达成一致的意见，最小化差异。

2．通过测试来检验软件的功能是否满足用户需求，对出现问题的地方及时的完善和修改。