**专业方向综合项目 项目需求规约**

**张尧 陈东 姜其升**

**TEam 12**

**MEETING！**

基于微信小程序的协同日程安排应用

**目录**

[1. 介绍 2](#_Toc44760974)

[1.1. 目的 2](#_Toc44760975)

[1.2. 预期读者 2](#_Toc44760976)

[1.3. 项目范围 2](#_Toc44760977)

[2. 总体描述 3](#_Toc44760978)

[2.1. 项目特色 3](#_Toc44760979)

[2.2. 用户类型和特征 3](#_Toc44760980)

[2.3. 运行环境 3](#_Toc44760981)

[3. 系统与其他系统的接口 4](#_Toc44760982)

[3.1. 用户接口 4](#_Toc44760983)

[3.2. 软件接口 4](#_Toc44760984)

[3.3. 硬件接口 4](#_Toc44760985)

[3.4. 通信接口 4](#_Toc44760986)

[4. 主要功能需求描述 5](#_Toc44760987)

[4.1. 文字描述 5](#_Toc44760988)

[4.2. 用例图 6](#_Toc44760989)

[4.2.1. 系统整体用例图 6](#_Toc44760990)

[4.2.2. 详细用例说明 6](#_Toc44760991)

[5. 非功能需求描述 14](#_Toc44760992)

[5.1. 性能需求 14](#_Toc44760993)

[5.2. 安全性需求 14](#_Toc44760994)

[5.3. 兼容性需求 15](#_Toc44760995)

[5.4. 可靠性需求 15](#_Toc44760996)

[5.5. 易用性 15](#_Toc44760997)

# 介绍

## 目的

Meeting系统（以下称本系统）主要为了解决当今开会时的痛点：当一个小组中的所有成员需要敲定一个开会时间时，会进行大量费时费力的讨论。本系统将提供一个日程管理的功能，使小组成员间能够共享显示每个人的日程安排，并根据所有小组成员的空闲时间自动推荐开会时间，实现快速的开会或其他活动时间确定。

为明确本系统的需求、安排实施规划与进度、组织软件开发与测试，撰写本文档。  本文档的最终审核版本将作为后续的设计、开发以及测试的参照标准。

## 预期读者

本文档的预期读者为：项目经理、系统架构师、系统管理员、程序员、最终用户、质保人员。

## 项目范围

本系统将在提供日程管理的功能基础上，提供小组事件的管理。以小组中所有成员的日程信息为基本数据，为组长快速确定小组事件时间提供便捷。

主要的功能有：

1. 个人事件的添加与管理
2. 工作组的创建、退出
3. 通过微信群的邀请自动加入同一个工作组
4. 查看工作组中他人日程
5. 根据工作组成员空闲时间及事件优先级自动推荐公共事件时间
6. 勿扰模式

本系统将业务范围集中在一周内的个人日程及小组事件管理上，尽最大可能将功能集中并考虑详尽，主要提供的核心功能为工作组中共享事件的自动推荐和创建、管理，辅以小组成员及权限的管理、个人事件的管理等功能。而将不会涉及到超过一周的事件管理、万年历查看功能等与核心功能无关的功能。

本系统的最终目标是提供一个能够便捷地通过微信小程序和微信群的良好联动生态，提供给用户管理个人短期内日程、便捷创建工作组并快速约定集体事件时间的应用，增加小组合作或活动策划的效率。

# 总体描述

## 项目特色

不同于其他日程管理软件，本系统在个人日程管理的基础上，创新地引入了工作组的概念，每个用户可以作为组长创建工作组并通过分享到微信群，邀请微信群中的小组成员，也可以作为成员加入到多个工作组中。

在工作组中，每个成员可以查看小组其他成员的日程信息，当然为了隐私考虑，可以在每个工作组中设置他人是否可以查看事件的详细描述。

除此之外，我们为每个日程增添了优先级，并辅以不同的颜色区分显示。

本系统最核心也是最具特色的功能是：在工作组中快速创建公共事件。当组长要创建一个公共事件时，他将选择公共事件的持续时间及优先级，系统则将自动地筛选所有小组成员的空闲时间进行时间推荐。如果没有合适的空闲时间，系统也将通过覆盖掉低优先级的事件进行公共事件时间确定。通过这样的每个人主动设置-被动反馈的方式，系统实现了公共事件时间快速设定，为工作增加了效率。

## 用户类型和特征

本系统的用户可能包括：

1. 需要进行大量组队项目工作的大学生群体
2. 需要商定集体活动时间的群组成员
3. 需要进行简单日程管理的所有人

本系统用户的特征为：

需要一个系统来进行自己短期内（7日）的日程管理。除此之外，在一些场景下，需要进行多人参与的小组活动。这些活动需要确认所有或大部分小组成员的空闲时间信息才能敲定时间，而本身的讨论又十分费时费力。

## 运行环境

系统运行环境为：

前端：WeChat Mini Program平台

服务层：Ubuntu 16.04

Java 1.8

数据库：Ubuntu 16.04

MySQL 5.6.40

# 系统与其他系统的接口

## 用户接口

用户可使用智能移动设备通过在微信小程序平台搜索“Meeting”进入本系统的用户界面，通过在设备上的点击、滑动、长按、倾斜与系统进行交互。

## 软件接口

本系统后端环境需要运行在Windows 8及以上、MacOS 10.7及以上或Ubuntu 14.04及以上并装有Java Development Kits 1.8的操作系统之上。

本系统前端环境需要运行在WeChat 7.0.0以上，基于Mobile Chrome 67渲染。

在系统的服务端与MySQL数据库的连接和操作上，我们主要使用MyBatis框架。它支持定制化 SQL、存储过程以及高级映射。相较于JDBC，MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集。

## 硬件接口

访问本系统前端微信小程序，需要能够运行微信小程序的智能移动设备（如手机、平板电脑等），且微信具有访问设备陀螺仪的权限。

本系统后端运行需要计算机硬件要求内存大小至少500M，硬盘64G以上。

## 通信接口

前端与后端进行的数据传输采用HTTPS协议，它是由SSL+HTTP协议构建的可进行加密传输、身份认证的网络协议，比HTTP协议更加安全。

# 主要功能需求描述

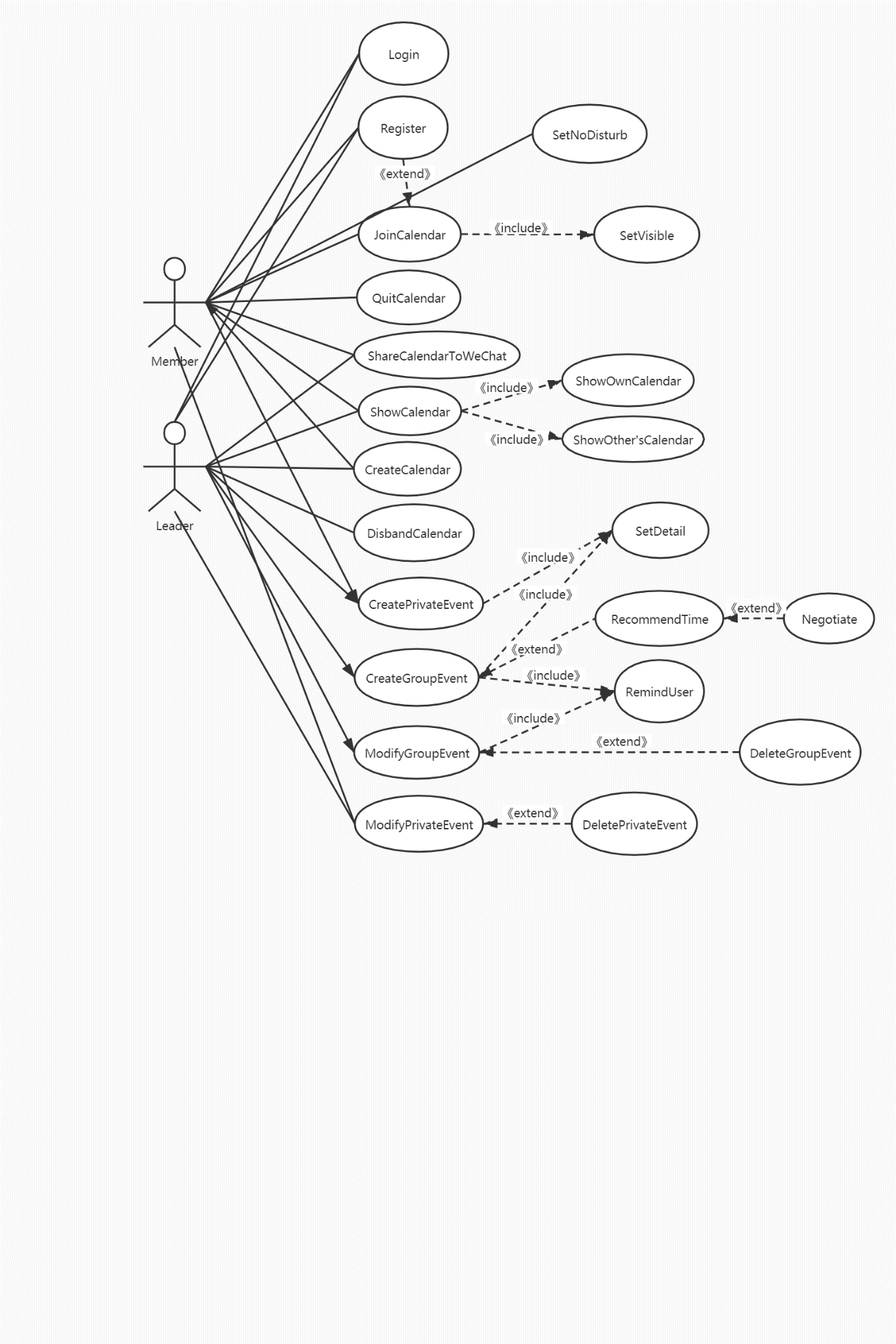
## 文字描述

本系统想要实现的功能可以大致归类为基于个人日程表的功能和基于团队日程表的功能。在个人方面，允许用户在日程表上进行个人事件的添加、修改和删除。在团队管理方面，允许用户创建团队、分享团队、加入团队，组员和组长还可以分别退出团队和解散团队。在团队日程管理方面，允许组长进行团队事件的创建、修改和删除，组员则可以对某组开启勿扰模式，在该模式下，团队事件不会被添加到个人日程中。

除此之外，本系统还有一些附加功能：如在加入时可以选择是否对某组开启隐私模式，在该模式下个人事件的名称不会被他人看到；在创建团队事件时，本系统还可以自动寻找团队的公共空闲时间并给出推荐时间段，方便组长更好地组织团队。

## 用例图

### 系统整体用例图



### 详细用例说明

|  |  |
| --- | --- |
| 用例名称 | 创建个人事件 |
| 标识符 | UC01 |
| 主要参与者 | 小组成员或组长，两者在创建个人事件的时候没有差别 |
| 说明 | 用户在不产生时间冲突的前提下在日程表上创建一个个人事件 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中 |
| 后置条件 | 1、用户输入的所有信息被接收，该事件被添加到个人日程中 |
| 被包含的用例 | 设置事件详细信息 |
| 触发器 | 用户点击添加个人事件 |
| 基本操作流程 | 1. 设置事件详细信息   1.1、输入事件的日期、开始时间和结束时间  1.2、输入事件的优先级  1.3、选择事件是否重复及是否添加事件详细描述  2、提交事件信息 |
| 可选操作流程 | 1. 事件名称为空：弹出提示，用户补充事件名称 2. 事件时间段冲突：弹出提示，用户重新输入事件的开始时间和结束时间 3. 未指定时间优先级：弹出提示，用户补充事件优先级 |
| 优先级 | 必须实现 |
| 使用频率 | 每天多次 |
| 使用方式 | 通过日程表界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 修改个人事件 |
| 标识符 | UC02 |
| 主要参与者 | 小组成员或组长，两者在修改个人事件的时候没有差别 |
| 说明 | 用户选择一个事件并对其基本信息进行修改或删除 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中 3. 用户的日程表上至少存在一个个人事件 |
| 后置条件 | 1、用户输入的修改要求被同步更新到日程表中 |
| 被扩展的用例 | 删除个人事件 |
| 触发器 | 用户长按某个个人事件 |
| 基本操作流程 | 1. 输入更新后事件的名称 2. 输入更新后事件的优先级   3、输入更新后事件的重复信息和详细描述  4、提交修改信息 |
| 可选操作流程 | 1. 用户选择删除事件：弹出提示，将该事件从用户日程表中移除   2、修改后事件名为空：弹出提示，用户重新输入事件的名称 |
| 优先级 | 必须实现 |
| 使用频率 | 每天多次 |
| 使用方式 | 通过日程表界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 创建团队事件 |
| 标识符 | UC03 |
| 主要参与者 | 组长，只有组长拥有创建团队事件的权力 |
| 说明 | 组长在不产生时间冲突的情况下创建一个需要整个团队参与的事件 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中   3、用户在当前组中的身份是组长 |
| 后置条件 | 1、用户输入团队事件被添加到所有组员的日程表当中 |
| 被包含的用例 | 设置事件详细信息 |
| 被扩展的用例 | 时间段推荐 |
| 触发器 | 用户点击添加组事件按钮 |
| 基本操作流程 | 1、设置事件详细信息  1.1、输入事件的名称、日期、开始时间和结束时间  1.2、输入事件的优先级  1.3、选择事件是否重复及是否添加事件详细描述  2、提交事件信息 |
| 可选操作流程 | 1. 用户选择时间段推荐    1. 用户输入想要添加的团队事件长度    2. 系统返回三个不冲突的事件段    3. 用户选择其中之一，系统跳转到添加界面并自动填充好时间 2. 事件名称为空：弹出提示，用户补充事件名称 3. 事件时间段冲突：弹出提示，用户重新输入事件的开始时间和结束时间   4、未指定时间优先级：弹出提示，用户补充事件优先级 |
|  | 必须实现 |
| 使用频率 | 取决于团队事件密度 |
| 使用方式 | 通过团队日程表界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 修改团队事件 |
| 标识符 | UC04 |
| 主要参与者 | 组长，只有组长拥有修改团队事件的权力 |
| 说明 | 组长在不产生时间冲突的情况下修改一个需要整个团队参与的事件 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中 3. 用户在当前组中的身份是组长 4. 该组至少存在一个团队事件 |
| 后置条件 | 1、用户的修改要求被同步到所有组员的日程表当中 |
| 被扩展的用例 | 删除团队事件 |
| 触发器 | 用户长按某个团队事件 |
| 基本操作流程 | 1. 输入更新后事件的名称 2. 输入更新后事件的优先级   3、输入更新后事件的重复信息和详细描述  4、提交修改信息 |
| 可选操作流程 | 1. 用户选择删除事件：弹出提示，将该事件从用户日程表中移除   2、修改后事件名为空：弹出提示，用户重新输入事件的名称 |
| 优先级 | 必须实现 |
| 使用频率 | 取决于团队事件修改密度 |
| 使用方式 | 通过团队日程表界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 创建团队日历 |
| 标识符 | UC05 |
| 主要参与者 | 组长或成员，两者在创建新团队时是平等的 |
| 说明 | 用户创建一个新团队 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中 |
| 后置条件 | 1、用户新建的日历被同步到组列表中，该用户成为该组的组长 |
| 触发器 | 用户点击新建日历按钮 |
| 基本操作流程 | 1. 用户输入新建的组名称 2. 点击提交按钮 |
| 可选操作流程 | 1、输入的组名为空：给出提示，用户补充输入组名称 |
| 优先级 | 必须实现 |
| 使用频率 | 每月多次 |
| 使用方式 | 通过团队管理界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 解散团队日历 |
| 标识符 | UC06 |
| 主要参与者 | 组长，只有组长拥有解散团队的权力 |
| 说明 | 组长解散一个现有的团队并清除组内所有相关信息 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中 3. 用户在当前组中的身份是组长 |
| 后置条件 | 1. 该组从组长的管理列表和组员的参加列表中被同时移除 2. 该组内现存和曾有的所有团队事件被同步删除 |
| 触发器 | 组长点击组对应的删除按钮 |
| 基本操作流程 | 1. 给出删除警告提示 2. 用户确认删除 |
| 可选操作流程 | 1、用户取消删除操作：退出用例 |
| 优先级 | 必须实现 |
| 使用频率 | 使用频率较低，每月1-2次 |
| 使用方式 | 通过团队管理界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 分享团队日历 |
| 标识符 | UC07 |
| 主要参与者 | 组长或组员 |
| 说明 | 用户将某个自己加入/创建组的链接分享到微信聊天中邀请他人加入 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中 |
| 后置条件 | 1、分享链接被成功转发到指定微信聊天中 |
| 触发器 | 点击组对应的分享按钮 |
| 基本操作流程 | 1. 给出分享提示 2. 用户确认分享 |
| 可选操作流程 | 1、用户取消分享操作：退出用例 |
| 优先级 | 必须实现 |
| 使用频率 | 每周多次 |
| 使用方式 | 通过团队管理界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 加入团队日历 |
| 标识符 | UC08 |
| 主要参与者 | 组长或组员 |
| 说明 | 用户点击他人分享的链接加入一个组 |
| 前置条件 | 1. 用户处在良好的网络环境中 |
| 后置条件 | 1、该用户被成功加入链接指定的团队日历 |
| 被包含的用例 | 设置隐私模式 |
| 被扩展的用例 | 用户注册 |
| 触发器 | 点击聊天中其他用户的分享链接 |
| 基本操作流程 | 1. 给出加入确认信息 2. 用户选择是否对该组开启隐私模式 3. 点击确认加入按钮 |
| 可选操作流程 | 1、用户未注册：弹出访问微信个人信息请求，用户同意注册后继续给出确认信息，否则退出用例。 |
| 优先级 | 必须实现 |
| 使用频率 | 每月多次 |
| 使用方式 | 通过微信聊天界面的分享接口 |

|  |  |
| --- | --- |
| 用例名称 | 退出团队日历 |
| 标识符 | UC09 |
| 主要参与者 | 组员 |
| 说明 | 组员退出某个已加入的团队日历 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中 3. 用户指定的团队存在且身份为组员 |
| 后置条件 | 1. 指定团队日历从用户的加入列表中移除 2. 指定团队日历的团队事件从用户的日程表中移除 |
| 触发器 | 点击组对应的退出按钮 |
| 基本操作流程 | 1. 给出警告信息 2. 用户确认退出该组 |
| 可选操作流程 | 1、用户取消退出操作：退出用例 |
| 优先级 | 必须实现 |
| 使用频率 | 每月1-2次 |
| 使用方式 | 通过组管理界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 设置勿扰模式 |
| 标识符 | UC10 |
| 主要参与者 | 组员 |
| 说明 | 组员对某个组设置勿扰模式，勿扰模式开启后，团队事件将不会被添加到该成员的个人日程中 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中 3. 用户指定的团队存在且身份为组员 |
| 后置条件 | 1、用户在该组的勿扰模式信息被更新 |
| 触发器 | 点击组对应的勿扰按钮 |
| 基本操作流程 | 1. 给出警告信息 2. 点击确认按钮 |
| 可选操作流程 | 无 |
| 优先级 | 必须实现 |
| 使用频率 | 每月多次 |
| 使用方式 | 通过团队日程管理界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 查看日程安排 |
| 标识符 | UC11 |
| 主要参与者 | 组员或组长 |
| 说明 | 用户对所有的事件（包括私人事件和团队事件）进行查看 |
| 前置条件 | 1. 用户已经注册 2. 用户处在良好的网络环境中 |
| 后置条件 | 无 |
| 被包含的用例 | 查看个人日程、查看他人日程 |
| 触发器 | 用户点击个人日程安排按钮 |
| 基本操作流程 | 1、用户查看个人日程安排  2、用户切换组  3、用户切换组内成员，查看他人日程安排 |
| 可选操作流程 | 无 |
| 优先级 | 必须实现 |
| 使用频率 | 每天多次 |
| 使用方式 | 通过日程管理界面接口 |

|  |  |
| --- | --- |
| 用例名称 | 注册 |
| 标识符 | UC12 |
| 主要参与者 | 组员或组长 |
| 说明 | 初次进入系统的用户申请获得使用系统的权限 |
| 前置条件 | 1. 用户处在良好的网络环境中 |
| 后置条件 | 无 |
| 被包含的用例 | 无 |
| 触发器 | 用户初次进入本系统 |
| 基本操作流程 | 1、用户在微信小程序平台搜索或点击他人分享的链接初次进入本系统  2、系统提示用户注册  3、用户点击同意注册  4、系统将获取用户微信账户昵称、头像信息，作为本系统的账户信息，并创建新账户 |
| 可选操作流程 | 1、用户取消注册操作：退出用例 |
| 优先级 | 必须实现 |
| 使用频率 | 只有1次 |
| 使用方式 | 通过系统外部接口 |

Extend和include的用例以基本流程和可选流程的形式已经都出现在了上述表格中

# 非功能需求描述

## 性能需求

**响应时间：**

在99％的情况下，在网络畅通时，一般时段响应时间不超过1.5秒，高峰时段不超过3秒。

每次操作从点击到第一个界面显示出来所需要的时间不得超过1秒。

在推荐配置环境下：登录响应时间在0.5秒内，打开信息条目响应时间1秒内。

**业务量：**

估计用户数为500人，每天登录用户数为200左右。

系统可以同时满足100个用户请求，并为1000个并发用户提供浏览功能。

**系统容量：**

支持3000个用户，支持GB级数据。

数据库表行数不超过5万行，数据库最大容量不超过30GB，磁盘空间至少需要40G以上。

**精度：**

在计算推荐公共事件时间时，推荐时间段误差<0.5小时。

**资源使用率：**

对移动设备 CPU占用率<=50%。

内存占用率<=50%。

## 安全性需求

**权限访问控制**

用户在经过登录操作进行身份认证后，只能访问其权限范围内的数据，只能进行其权限范围内的操作。

不同的用户具有不同的身份和权限，需要在用户身份真实可信的前提下，提供可信的授权管理服务，保护数据不被非法/越权访问和篡改，要确保数据的机密性和完整性。

**日志控制**

提供运行日志管理及安全审计功能，可追踪系统的历史使用情况。

**恶意请求防御**

能经受来自互联网的一般性恶意攻击。如病毒（包括木马）攻击、口令猜测攻击、黑客入侵等。

## 兼容性需求

**系统与其他软件兼容性**

后端系统应支持MacOS，Windows，Ubuntu操作系统。

前端系统应兼容前后多个版本的微信软件。

**与操作系统交互**

最多只有5%的系统实现需要具体到特定的操作系统。

**可移植性**

移植系统时无需配置大量环境，

## 可靠性需求

**输入可靠性**

系统对输入字符有提示，数据有检查，防止数据异常造成的系统失效。

**系统健壮性**

系统健壮性强，应该能处理系统运行过程中出现的各种异常情况，如：人为操作错误、输入非法数据、硬件设备失败等，系统应该能正确的处理，恰当的回避。

**运行可靠性**

系统7x24小时运行，全年持续运行故障停运时间累计不能超过10小时。

系统缺陷率每1,000小时最多发生1次故障。

因软件系统的失效而造成不能完成业务的概率要小于5%。

## 易用性

**易理解性**

系统界面的图标及文字提示不应有歧义，需便于用户理解。

**易学习性**

绝大多数用户应该能在5分钟内根据系统所给的提示，了解系统的基本操作。

**易操作性**

系统界面组件的排布、按钮的设计及文字大小等应该符合界面友好性，满足HCI的设计规范。在进行页面布局时，除考虑美观外，也应考虑不同用户的操作习惯，用户进行的常用操作按钮放在易操作的位置。