

需求规格说明书

# 目录

[目录 1](#_Toc419142105)

[一. 引言 2](#_Toc419142106)

[1.1 编写目的 2](#_Toc419142107)

[1.2 项目背景 2](#_Toc419142108)

[1.3 参考资料 3](#_Toc419142109)

[二.需求规定 3](#_Toc419142110)

[2.1系统模块 3](#_Toc419142111)

[2.3基本设计概念和处理流程 4](#_Toc419142112)

[2.3.1 角色与业务流程 4](#_Toc419142113)

[2.3.2用例图 5](#_Toc419142114)

[2.4 用例描述 7](#_Toc419142115)

[2.4.1 登录 7](#_Toc419142116)

[2.4.2增加日程 7](#_Toc419142117)

[2.4.3修改日程 8](#_Toc419142118)

[2.4.4删除日程 9](#_Toc419142119)

[2.4.5延期 9](#_Toc419142120)

[2.4.6完成 10](#_Toc419142121)

[2.5 非功能的规定 10](#_Toc419142122)

[2.6 数据库管理能力要求 11](#_Toc419142123)

[2.7接口 11](#_Toc419142124)

# 一. 引言

## 1.1 编写目的

为了方便现代处于快节奏生活的人们合理安排时间，避免耽误事。尤其是在校大学生，每天学习生活和社团工作等忙得不可开交，此软件可以方便大学生进行日程管理。

本需求分析说明书的主要读者为（1）项目经理、设计人员和开发人员，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，规范化本系统的编写；（2）用户，是否满足用户需求，以期调整和完善该日程管理系统。

。

## 1.2 项目背景

随着我国经济的不断发展，生活节奏的加快，很多人感觉生活很忙以至于经常忘其他的事。

为了方便现代处于快节奏生活的人们合理安排时间，避免耽误事。尤其是在校大学生，每天学习生活和社团工作等忙得不可开交，合理安排时间及日程管理显得尤为重要。

此系统为了帮助人们及时、高效地完成自己的目标和任务，管理好日程。

## 1.3 参考资料

《构建之法——现代软件工程》 邹欣 著

《软件工程--实践者的研究方法》 RogerS.Pressman 著

课上PPT

# 二.需求规定

## 2.1系统模块

进入系统需要进行身份验证，软件功能主要分为新建日程，修改日程，删除日程，延期，完成这几个部分。新建日程输入日程信息，如主题，内容，地点，时间，开始时间，结束时间，优先级，备注等。修改则是对之前添加的日程进行修改。删除日程是因为变化删除之前已经添加的日程。延期则是如果之前添加的日程后推，进行延期。完成是选择相应日程改为完成

|  |  |  |  |
| --- | --- | --- | --- |
| **模块名称** | **功能** | **子功能** | **功能说明** |
| 登陆 | 用户登陆 | 日程管理 | 对日程进行操作 |
| 新增日程 | 增加日程 | 无 | 增加日程，输入相关信息 |
| 修改日程 | 客户修改日程 | 无 | 对之前的日程进行修改 |
| 删除日程 | 对添加的日程进行删除 | 无 | 对添加的日程进行删除 |
| 日程延期 | 对日程进行延期操作 | 无 | 对日程日期进行修改 |
| 完成 | 对完成的日程进行状态修改 | 无 | 对已完成日程进行修改 |

## 2.3基本设计概念和处理流程

### 2.3.1 角色与业务流程

此系统的基本角色为用户，用户业务流程图如图所示：

开始

是

结束

完成

删除日程

延期

修改日程

添加日程

否

登陆信息是否正确

登陆

### 2.3.2用例图

#### 系统总用例图：



#### 登录用例图



#### 新建日程



#### 修改日程

**删除日程**



#### 延期



#### 完成



## 2.4 用例描述

### 2.4.1 登录

|  |  |
| --- | --- |
| 用例名称： | 登录 |
| 用例ID： | USER-001 |
| 角色： | 用户 |
| 用例说明： | 用户登录系统 |
| 前置条件： | 输入用户名和密码 |
| 基本事件流： | 1. 用户登陆后显示已有的日程信息 2. 用户对日程进行操作 |
| 其它事件流： | 用户输入错误的用户名和密码则重新输入 |
| 异常事件流： |  |
| 后置条件： |  |

### 2.4.2增加日程

|  |  |
| --- | --- |
| 用例名称： | 增加日程 |
| 用例ID： | USER-002 |
| 角色： | 用户 |
| 用例说明： | 用户进行新建日程 |
| 前置条件： | 已经登陆成功 |
| 基本事件流： | 1. 添加新的日程信息 2. 保存信息 |
| 其它事件流： |  |
| 异常事件流： |  |
| 后置条件： |  |

### 2.4.3修改日程

|  |  |
| --- | --- |
| 用例名称： | 修改日程 |
| 用例ID： | USER-003 |
| 角色： | 用户 |
| 用例说明： | 对已有日程信息进行修改 |
| 前置条件： | 已经登陆成功 |
| 基本事件流： | 1. 对已有日程信息进行修改 2. 保存修改 |
| 其它事件流： |  |
| 异常事件流： |  |
| 后置条件： |  |

### 2.4.4删除日程

|  |  |
| --- | --- |
| 用例名称： | 删除日程 |
| 用例ID： | USER-004 |
| 角色： | 用户 |
| 用例说明： | 对已添加日程进行删除 |
| 前置条件： | 已经登陆 |
| 基本事件流： | 1.选择要删除日程  2.选择删除  3.确认删除 |
| 其它事件流： |  |
| 异常事件流： |  |
| 后置条件： |  |

### 2.4.5延期

|  |  |
| --- | --- |
| 用例名称： | 延期 |
| 用例ID： | USER-005 |
| 角色： | 用户 |
| 用例说明： | 对之前的日程进行延期 |
| 前置条件： | 已经登陆 |
| 基本事件流： | 1. 客户选择要延期的日程 2. 选择延长的日期； 3. 保存延期 |
| 其它事件流： |  |
| 异常事件流： |  |
| 后置条件： |  |

### 2.4.6完成

|  |  |
| --- | --- |
| 用例名称： | 完成 |
| 用例ID： | USER-006 |
| 角色： | 用户 |
| 用例说明： | 对已完成日程进行完成状态更改 |
| 前置条件： | 用户登陆并且已完成日程 |
| 基本事件流： | 1. 选择完成的日程 2. 保存 |
| 其它事件流： |  |
| 异常事件流： |  |
| 后置条件： |  |

## 2.5 非功能的规定

 （1）系统健壮：具有一定的容错能力，并且以友好的方式告之用户

1. 可复用性：系统组件可重复使用
2. 可扩展性：符合开闭原则，添加新功能时不会对原系统造成太大影响
3. 高效性：数据库设计良好，能够与系统较高效的交互
4. 安全性：系统可自动备份数据库，防止意外操作引起的数据损坏



## 2.6 数据库管理能力要求

本系统使用SQL Server数据库，用来存储订单，点餐单，结账单，商品等系统信息，数据库要求如下：

1. 在不影响效率的情况下尽可能满足3N范式
2. 数据库表之间联系紧密，便于维护
3. 满足数据库设计的重要原则

## 2.7接口

(1) 用户接口

用户在登录页面输入的用户名和密码，系统根据用户名判断用户的身份并赋予相应的权限。登录系统后即可进行相应的操作。

(2) 硬件接口（逻辑结构，物理地址）

软件支持常用的服务器及个人计算机。

(3) 软件接口

软件运行在windows操作系统之上，系统分为三层结构：表示层、应用层、数据存储层，采用MVC架构，软件在后台与数据库进行交互。

(4) 通信接口（局域网，远程串行设备）

系统暂时不作考虑。