

**G06小组事务提醒系统**

**总体设计**

****

**小组编号： G6**

**组 长： 江 泓 31501300**

**组 员： 曾雨晴 31501356**

**于欣汝 31501359**

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ √ ]草稿  [ ] 正式发布  [ ] 正在修改 | 文件标识： | G06-project |
| 当前版本： | V1.1.0 |
| 作者： | 江泓 曾雨晴 于欣汝 |
| 完成日期： | 2017年5月11号 |

**2017年5月11日**

**版本历史**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本/状态 | 参与者 | 起止日期 | 备注 |
| **V1.0.0** | 江泓 曾雨晴 于欣汝 | **2017.4.29** | **写好总体设计大致框架** |
| **V1.0.1** | 曾雨晴 | **2017年5月7日** | **添加其他方案的支持及补充** |
| **V1.1.0** | 江泓 曾雨晴 于欣汝 | **2017年5月11日** | **修改部分图表** |

目录

[一、引言 4](#_Toc20414)

[1、编写目的 4](#_Toc28974)

[2、背景 4](#_Toc2842)

[3、参考资料 4](#_Toc27021)

[二、总体设计 4](#_Toc5306)

[1、需求规定 4](#_Toc7702)

[2、运行环境 5](#_Toc1550)

[3、基本设计概念和处理流图 5](#_Toc5105)

[a)系统数据流图 5](#_Toc25362)

[b)状态转换图 6](#_Toc27843)

[4、软件结构 6](#_Toc1572)

[5、HIPO图 7](#_Toc910)

[6、功能需求与程序的关系 7](#_Toc10204)

[三、接口设计 7](#_Toc14787)

[1、用户接口 7](#_Toc19018)

[2、外部接口 8](#_Toc15855)

[3、内部接口 8](#_Toc19335)

[四、运行设计接口 8](#_Toc8344)

[1、运行模块组合 8](#_Toc16773)

[2、运行控制 9](#_Toc29008)

[3、运行时间 9](#_Toc4463)

[五、系统论据结构设计 9](#_Toc11521)

[1、逻辑结构设计要点 9](#_Toc1834)

[a)数据库表结构如下 9](#_Toc20482)

[b)系统ER图结构如下 9](#_Toc5551)

[2、物理结构设计要点 9](#_Toc12667)

[3、数据结构与程序的关系 11](#_Toc28497)

[六、系统出错处理设计 11](#_Toc15121)

[1、出错信息 11](#_Toc23862)

[2、补救措施 11](#_Toc26374)

[七、 其他方案设计 11](#_Toc14215)

[1、概述 11](#_Toc37)

[2、拓展 12](#_Toc11294)

# 一、引言

## 1、编写目的

本文档旨在为事务备忘系统需求说明书提出概要设计，为编程起指导作用，包括处理流程、结构与模块的划分以及接口设计。预期读者为开发人员和指导人员

## 2、背景

软件名称：事务备忘系统

任务提出者：G06小组

开发者：G06小组

用户：长时间在电脑前的人

## 3、参考资料

《软件工程导论》——清华出版社

# 二、总体设计

## 1、需求规定

本软件主要是为了帮助长时间在电脑前的人合理安排时间而开发的一款备忘录系统，它应满足对事务的添加、删除、修改、统计、提醒这五个核心功能，采用C/S架构，在电脑上对用户进行提醒。当用户开启软件后，到预定时间时，提醒用户。

此外，还应满足以下几点：

1）每天的事务总量应当没有上限，总事务总量应当没有上限。

2）删除的事务将被放入回收站内，保留该事务创建的日期与内容，并且有从回收站移除功能。

3）事务根据重要性分两栏添加和显示

4）当创建与删除事务时，列表应当自动排序，以时间为基准，从早到晚排列。

5）若创建时有在同一个时间有事务冲突，则提醒有时间冲突。

6）当到了事务的提醒时间时，提醒用户

7）事务有完成和未完成状态，分已完成和未完成两个列表展现。

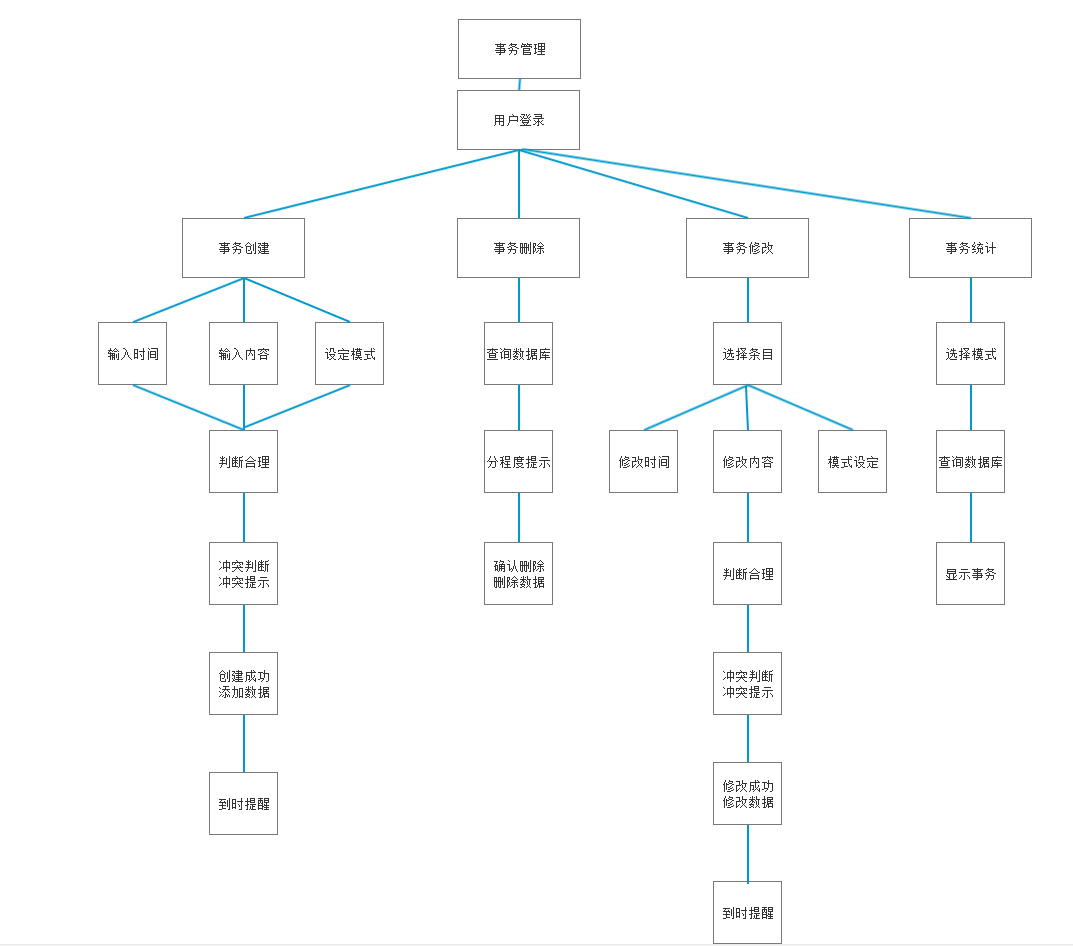
统计事务应当可以自由选定时间周期，最长为1年，最短为1分钟。对统计好的事务应当有已完成和未完成这两项的划分

## 2、运行环境

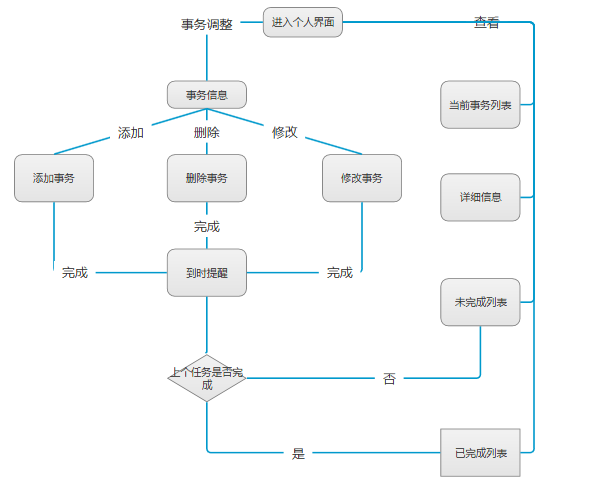
PC端windows10

## 3、基本设计概念和处理流图

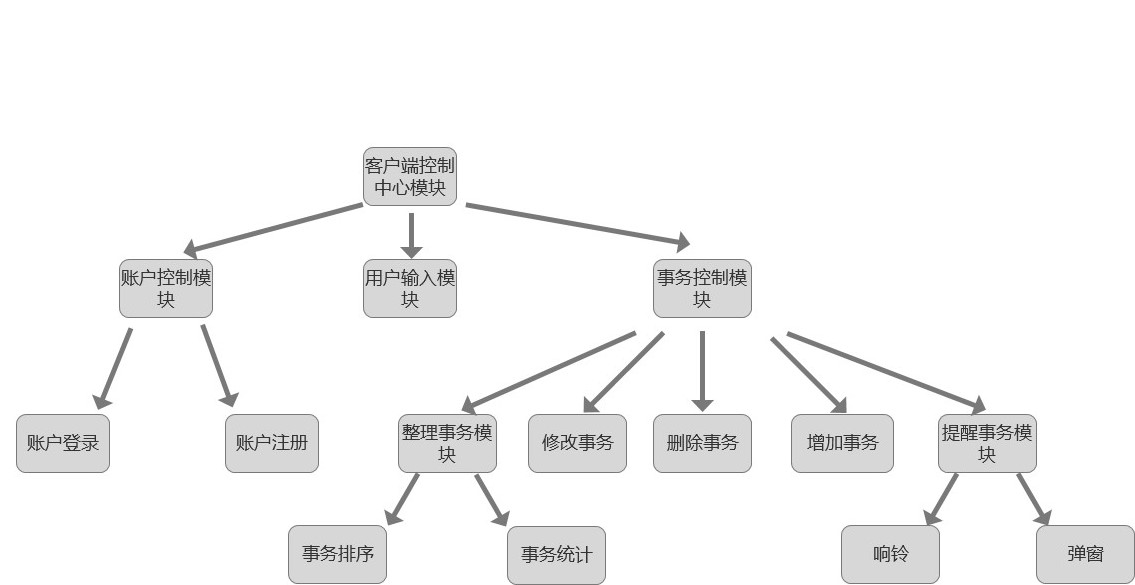
### a)系统数据流图



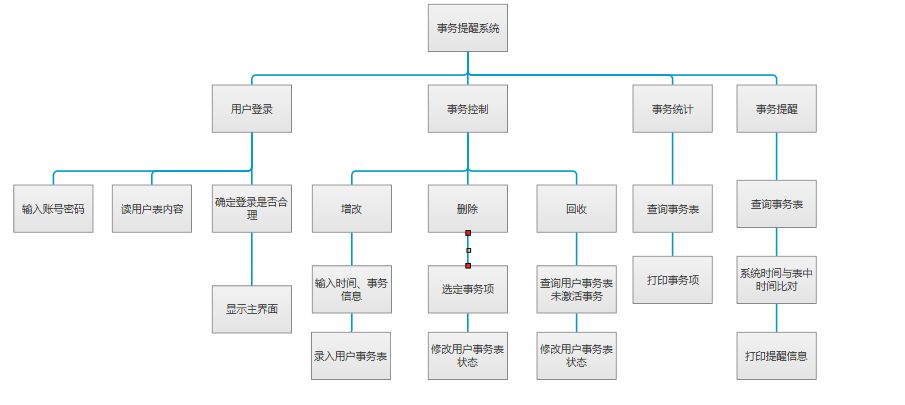
### b)状态转换图



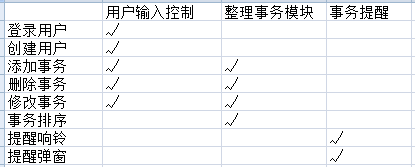
## 4、软件结构



## HIPO图



## 6、功能需求与程序的关系



# 三、接口设计

## 1、用户接口

采用图形用户界面：

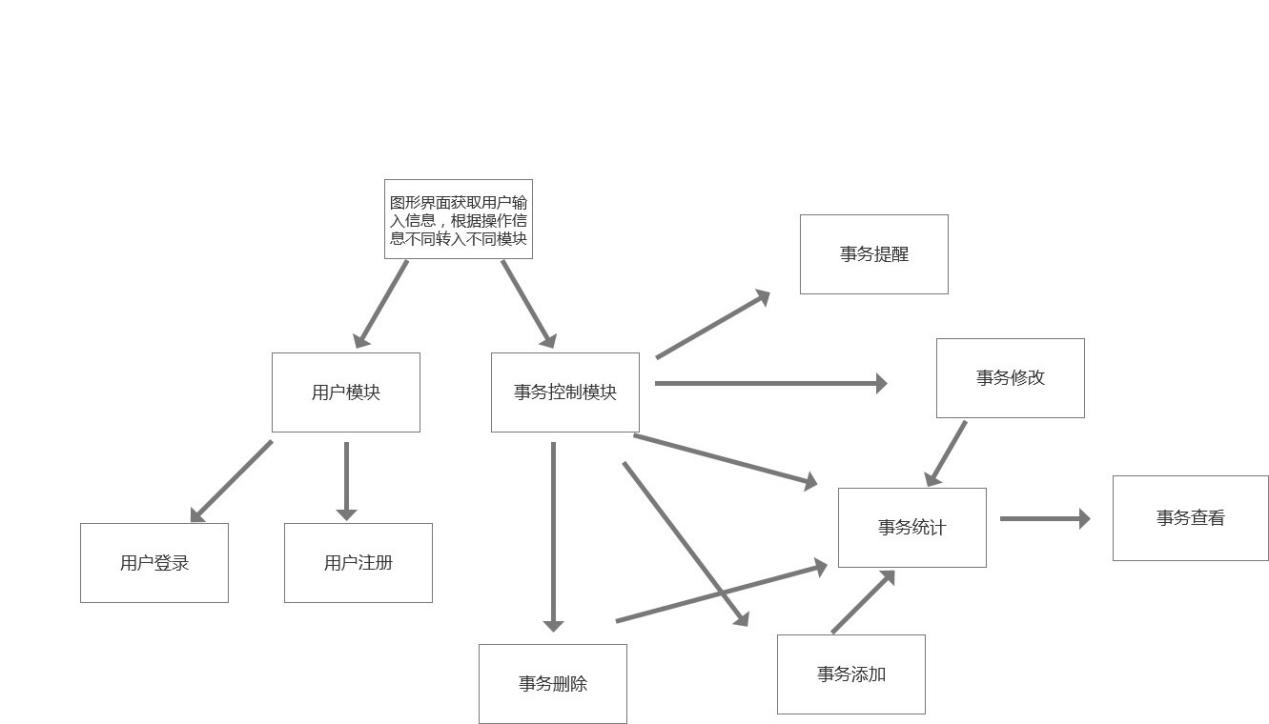
1. 登录界面：输入用户名和密码，选择登录或者注册
2. 事务添加：输入事务名称，选择提醒时间，备注
3. 事务修改：对事务的时间、内容进行修改
4. 事务提醒：显示当前时间，事务内容
5. 查看事务：显示已有事务，按时间排序。有未完成事务的统计

## 2、外部接口

无外部接口

## 3、内部接口

图形界面获取用户的各项输入，根据不同操作请求进入相应的操作模块。操作模块如下：



# 四、运行设计接口

## 1、运行模块组合

系统将初步分为

1. 用户登录模块
2. 事务增删改模块
3. 声音提醒模块
4. 事务排序模块

总共四个模块

其中初步确定

组长江泓将负责ac两个模块，

小组组员曾雨晴负责d模块，

小组组员于欣汝负责b模块。

## 2、运行控制

## 3、运行时间

本系统体积小，功能简单，不需要联网，因此运行时间不成问题

# 五、系统论据结构设计

## 1、逻辑结构设计要点

### a)数据库表结构如下

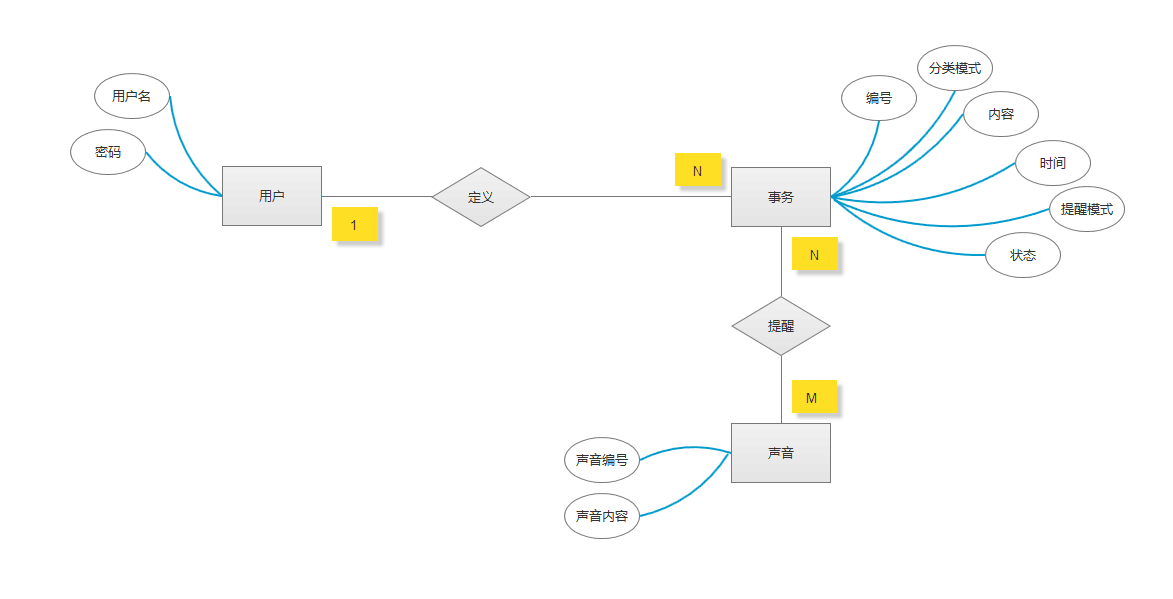
用户表：用户名（key），用户密码

事务表：事务编号（key），事务内容，提醒时间，状态，提醒模式，分类模式

声音表：声音编号（key），声音内容

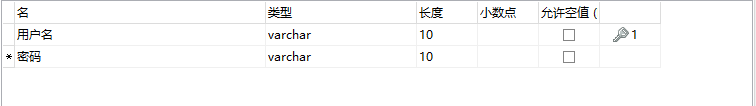
用户事务表：用户名（外码），事务编号（外码），事务内容，提醒时间，状态，分类模式

### b)系统ER图结构如下

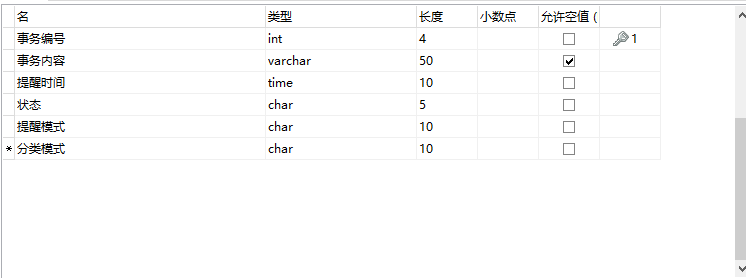


## 2、物理结构设计要点

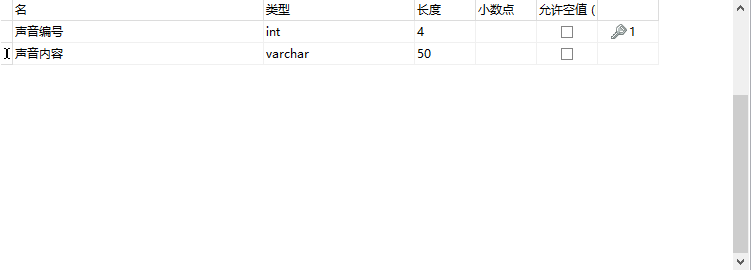
用户表：



事务表：



声音表：



用户事务表：



## 3、数据结构与程序的关系

# 六、系统出错处理设计

## 1、出错信息

事务时间冲突：已存在一个与新创建或修改的事务时间相同的事务

## 2、补救措施

建议用户修改时间或者将两个事务合并

# 其他方案设计

## 1、概述

在以上的叙述中，我们小组将采用C/S架构，开发客户端软件，用户通过下载软件来获取软件的使用权，但考虑到C/S架构下的一些不便以及不合理之处。我们小组推出第二种实施方案。在第二方案中，我们将采用B/S架构，来建设一个网页，用户通过访问网页就可以做出一系列对事务的操作，提高了用户操作的方便性。在该方案中，我们将依旧利用上述的技术，只是在实现的时候改为网页架设。

## 2、拓展

但在考虑到技术方面的需求时，我们小组组员并没有制作网页的经验，也对网页的管理十分陌生，因此我们小组将B/S架构下的事务备忘系统作为一种参考的实施方案，将在未来的几个工作日内，结合技术和经济可行性以及对用户需求的再确认，来考虑是否要将备用方案升级为第一方案，当然因为在C/S架构下和B/S架构下，我们的软件大多部分的设计都是一致的，所以我认为这不失为一个很好的选择。