**记账系统设计说明书**

1.引言

1.1报告撰写的目的

个人记账系统是针对希望有更方便快捷记账手段的客户设计，是可以完成在线记账保存，限额设定，余额计算，甚至是多用户共享一个账本的系统。

此可行性研究报告，是为实现在最短的时间内以最小的代价确定设计个人记账系统的问题是否可以解决，从而确定进一步对系统进行开发的流程步骤。

1.2项目背景

今年来随着信息技术的进步，理财观念的提升，电子记账越来越流行，电子记账同步的方式更加方便快捷，有很大需求。

1.3项目目的

以网站形式提供记账服务，提高记账准确率和效率

2.技术可行性

2.1可行性因素

a.项目的效率远大于成本；

b.技术可行，以现有技术完全可以承担起开发的任务；

c.操作可行，软件简单易用

2.2技术条件

系统：Windows10

开发平台：pycharm

开发语言：python

架构：B/S

服务器：Tomcat6.0

前台：ExtJs/Html/Javascript/Css

后台：Structs2（JDK）

数据库：MySQL

版本控制：github

2.3.系统流程图

账目数据库

账目管理：

设置上限，设置时间，记录每笔账目

浏览器

用户

3.系统层次结构图

4.概要设计说明书

4.1.总体设计

4.1.1

用户注册及登录

注册新用户，已有用户登录

4.1.2

账目信息查询

登录成功后，查询历史账目记录

4.1.3

账目信息记录

登录成功后，可以设置消费额度，消费时间，记录每笔账目

4.2功能模块清单

|  |  |  |
| --- | --- | --- |
| 记账系统 | | |
| 序号 | 功能 | 说明 |
| 1 | 查询 | 实现用户查询历史记录 |
| 2 | 记录 | 实现用户记录账目 |

4.3数据结构设计

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 数据类型 | 约束条件 | 注释 |
| 登录名 | Varchar(10) | Primary key, not null |  |
| 密码 | Varchar(6) | Not null |  |
| 基本设定 | Data struct {  Datetimes（start,end）  Int num  } | Not null | Date time start  为记账开始时间  Date time end  为记账截止时间  Int num设置消费额度 |
| 消费记录 | Data struct{  Varchar/用户名/  datetime/消费时间/  int/消费种类/  int/消费数目/  } | Not null  消费种类0-5 | 消费种类  0：吃；1：穿；2：住；3：用；4：行；5：娱乐服务 |

5详细设计说明书

网站设计：1.登录注册页面2.基本信息页面3.设置页面4.记录页面5.查询

登录注册：

1.注册：

1Register(){

2Account=input(用户账号)

3search(account)

4if(success)

5goto 2

6else

7Passward=Input(用户密码)

8Passward1=Input(再次输入密码)

9If(密码合理and passward=passward1)

10output注册成功

11else

12output 注册失败

13goto7}

2.登录

1Login{

2account=Input(账户名)

3search(account)

4 if(success)

5passward=input(密码)

6Else

7Goto 2

8judge(account,passward)

9if (success)

10output登录成功

11else goto5

}

基本信息：

Basicinfo{

Output(用户名)

Search(account)

If(当前时间不在datetime 范围or datetime 不存在)

Output(未设置时间)

Else

temp =0,遍历账目

if(账目datetime in datetimes) temp+=消费数目 //加总时间内的账目

if(temp>额度num)

output(当前已消费超额)

}

记录页面

Record{

While true

Input(消费种类，消费数目,消费时间)

newitem(账户=account,时间,种类,数目) //建立新账目

if(退出)

break

}

查询

Inquiry

{

Input(分类，时间)

If none

Search all

Search(账户，分类，时间)

Output() //按照时间分类输出账目

}