员工管理系统项目文档

作者：石磊、代朝阳 日期：2020.8.15

一、通讯结构体

typedef struct {

int id; /\*用户/员工ID号(唯一) \*/

char name[20]; /\*员工姓名 \*/

char username[20]; /\*用户名 \*/

char passwd[20]; /\*用户密码 \*/

int flags[2]; /\*返回信号 \*/

char sex[10]; /\*员工性别 \*/

int age; /\*员工年龄 \*/

char phone[11]; /\*员工电话 \*/

char addr[50]; /\*员工地址 \*/

int type[2]; /\*员工权限 \*/

}MSG;

其中，ID号是唯一的，用来进行身份识别，用户密码修改，员工信息增删改查都是基于ID号实现的。

二、返回信号

#define REGISTER 1 /\*注册 \*/

#define LOGIN 2 /\*登录 \*/

#define FIND 3 /\*忘记密码 \*/

#define QUIT 15 /\*退出 \*/

#define REG\_SUC 4 /\*注册成功\*/

#define LOGIN\_SUC 5 /\*登录成功\*/

#define CHECK\_OK 14 /\*校验成功\*/

#define ADD\_USER 6 /\*添加用户\*/

#define DELE\_USER 7 /\*删除用户\*/

#define CGE\_USER 8 /\*修改用户\*/

#define INQ\_INFO 9 /\*查看员工信息\*/

#define DELE\_SUC 11 /\*删除用户成功\*/

#define CGE\_SUC 12 /\*修改用户成功\*/

#define CGE\_PSW 13 /\*修改密码\*/

#define ERROR 66 /\*执行失败\*/

说明：返回信号用于识别通讯双方的意图。

其中，REGISTER、LOGIN、FIND、QUIT、ADD\_USER、DELE\_USER、CGE\_USER、INQ\_INFO、CGE\_PSW为客户端请求。

REG\_SUC、LOGIN\_SUC 、CHECK\_OK、DELE\_SUC、CGE\_SUC为执行成功后，返回给客户端的反馈信号。ERROR 为执行失败信号。

三、服务器功能介绍

1、主函数和程序流程

主函数处理监听到的客户端的链接请求。Accept函数接受请求，开始双方的信息通讯。本次项目采用TCP协议进行通讯。

当通讯双方建立好链接以后，产生新的套接字，通过这个新的套接字进行消息的收发。

链接成功后，接收客户端请求，判断请求意图（注册/登录/忘记密码/退出）。

当请求信息为“注册”时，调用usr\_register函数，判断账号是否存在，不存在则创建。并根据用户名赋予账号权限（管理员/普通用户）。

当请求信息为“忘记密码”时，调用passwd\_find函数，判断用户身份，认证成功则更新密码。

当请求信息为“登录”时，调用user\_login判断用户名、密码是否正确，正确则调用login\_win函数，进入二级界面。

进入二级界面后，服务器阻塞等待客户端请求。当客户端发送请求时，判断请求意图（添加员工/更新员工信息/查看员工信息/删除员工/修改密码/退出）。

当请求“添加员工”时，调用usr\_register函数，添加员工信息到数据库表中。

当请求“删除员工”时，调用emp\_remove函数，根据员工ID号从表中删除员工信息。

当请求“更新员工信息”时，调用emp\_data\_update函数，根据员工ID号更新员工信息。

当请求“查看员工信息”时，调用emp\_cat函数，根据员工ID号查看员工信息。

当请求“修改密码”时，调用passwd\_change函数，根据员工ID号修改密码。

当接收到“退出”指令时，退出login\_win函数循环。

2、功能函数介绍

监听处理函数

功能：监听客户端的链接请求

int listen\_socket(const unsigned int port,int backlog )

{

/\*创建套接字\*/

fd = socket(AF\_INET,SOCK\_STREAM,0);

/\*允许快速重用\*/

/\*填充地址信息结构体\*/

/\*绑定地址\*/

/\*监听\*/

if(0 > listen(fd,BACKLOG))

return fd;

}

一级界面中调用的函数：

1.注册处理函数

接收结构体中的信息。判断用户名、密码是否存在，不存在则将用户名和密码添加到 用户信息表 中，获取表中自动分配的ID作为员工ID号，将员工的ID、姓名、年龄、性别、手机号、地址依次填入到 员工信息表 中。将执行结果反馈给客户端。

void usr\_register(int newfd,MSG \*usrMsg)

{

memset(usrMsg,0,sizeof(MSG));

recv(newfd,usrMsg,sizeof(MSG),0);

/\*验证用户是否存在\*/

sprintf(sql,"select \* from usr\_data where username='%s' and passwd='%s' and type=%d;",\

usrMsg->username,usrMsg->passwd,\*usrMsg->type);

if(sqlite3\_get\_table(db,sql,&resultp,&nrow,&ncloumn,&errmsg) != SQLITE\_OK)

{ }

if(nrow == 0)

{

/\*添加用户名、密码到用户数据表中\*/

sprintf(sql,"insert into usr\_data values(null,'%s','%s',%d);",\

usrMsg->username,usrMsg->passwd,usrMsg->type[0]);

if(sqlite3\_exec(db,sql,NULL,NULL,&errmsg) != SQLITE\_OK)

{

printf("===== user register success =====\n");

/\*查找用户id\*/

sprintf(sql,"select id from usr\_data where username='%s' and passwd='%s' and type=%d;",\

usrMsg->username,usrMsg->passwd,\*usrMsg->type);

if(sqlite3\_get\_table(db,sql,&resultp,&nrow,&ncloumn,&errmsg) != SQLITE\_OK)

{

if(nrow >= 1)

{

usrMsg->id = atoi(resultp[1]);

printf(">>>your user id is >%d\n",usrMsg->id);

/\*添加员工信息\*/

sprintf(sql,"insert into emp\_data values(%d,'%s',%d,'%s','%s','%s');",\

usrMsg->id,usrMsg->name,usrMsg->age,usrMsg->sex,usrMsg->phone,usrMsg->addr);

if(sqlite3\_exec(db,sql,NULL,NULL,&errmsg) != SQLITE\_OK)

{

printf("===== employee message insert success.=====\n");

usrMsg->flags[0] = REG\_SUC;

send(newfd,usrMsg,sizeof(MSG),0);

}

}

}

}

}

}

2.忘记密码处理函数

接收结构体中的信息。根据用户ID判断用户名和用户ID是否匹配（此处简化了验证方式，可单独设置一个密保结构体成员），匹配成功，反馈给客户端，等待接收客户端的新密码，更新密码。将结果反馈给客户端。

void passwd\_find(int newfd,MSG \*usrMsg)

{

recv(newfd,usrMsg,sizeof(MSG),0);

sprintf(sql,"select \* from usr\_data where username='%s' and id=%d;",\

usrMsg->username,usrMsg->id);

if(sqlite3\_get\_table(db,sql,&resultp,&nrow,&ncloumn,&errmsg) != SQLITE\_OK)

{ }

if(nrow >= 1)

{

usrMsg->flags[0] = CHECK\_OK;

/\*将帐号存在结果发送给客户端\*/

send(newfd,usrMsg,sizeof(MSG),0);

recv(newfd,usrMsg,sizeof(MSG),0);

sprintf(sql,"update usr\_data set passwd='%s' where id=%d;",\

usrMsg->passwd,usrMsg->id);

if(sqlite3\_exec(db,sql,NULL,NULL,&errmsg) != SQLITE\_OK)

{

printf("===== password update success.=====\n");

usrMsg->flags[0] = CGE\_SUC;

send(newfd,usrMsg,sizeof(MSG),0);

}

}

}

3.登录处理函数

接收结构体中的信息。通过 用户信息表 验证用户名、密码是否存在，存在则登录成功并对用户权限进行设定，成功后调用登录成功函数进入二级界面。将用户权限和登录结果反馈给客户端。

void usr\_login(int newfd,MSG \*usrMsg)

{

recv(newfd,usrMsg,sizeof(MSG),0);

/\*验证用户是否存在\*/

sprintf(sql,"select \* from usr\_data where username='%s' and passwd='%s';",\

usrMsg->username,usrMsg->passwd);

if(sqlite3\_get\_table(db,sql,&resultp,&nrow,&ncloumn,&errmsg) != SQLITE\_OK)

{ }

/\*将权限信息发送给客户端\*/

is\_root(usrMsg);

if(nrow >= 1)

{

usrMsg->flags[0] = LOGIN\_SUC;

send(newfd,usrMsg,sizeof(MSG),0);

printf("===== usr login success.=====\n");

login\_win(newfd,usrMsg);

}

}

4.权限判断函数

将用户名与表中的用户名做对比，如果为“root”，判断其为管理员，否则判定为普通用户。

void is\_root(MSG \*usrMsg)

{

int ret = -1;

ret = strcmp("root",usrMsg->username);

if(ret == 0){

usrMsg->type[0] = 0;

}else{

usrMsg->type[0] = 1;

}

}

二级界面中调用的函数：

5.登录成功函数

接收结构体中的信息。判断请求类型，调用相应的功能处理函数。

void login\_win(int newfd,MSG \* usrMsg)

{

while(1){

recv(newfd,usrMsg,sizeof(MSG),0);

switch(usrMsg->flags[0])

{

case REGISTER:

usr\_register(newfd,usrMsg); /\*添加员工信息\*/

break;

case DELE\_USER:

emp\_remove(newfd,usrMsg); /\*删除员工信息\*/

break;

case INQ\_INFO:

emp\_cat(newfd,usrMsg); /\*查看员工信息\*/

break;

case CGE\_USER:

emp\_data\_update(newfd,usrMsg); /\*更新员工信息\*/

break;

case CGE\_SUC:

passwd\_change(newfd,usrMsg); /\*修改密码\*/

break;

case QUIT:

printf("back menu...\n");

return;

break;

default:

printf("make choice error!\n");

return;

break;

}

}

return;

}

6. 添加员工信息处理函数

接收结构体中的信息。调用注册函数。

7. 删除员工信息处理函数

接收结构体中的信息。根据员工ID判断员工是否存在，若存在，根据ID在表中删除该员工的所有信息。将删除结果反馈给客户端。

void emp\_remove(int newfd,MSG \*usrMsg)

{

/\*验证用户是否存在\*/

sprintf(sql,"select \* from usr\_data where id=%d;",usrMsg->id);

if(sqlite3\_get\_table(db,sql,&resultp,&nrow,&ncloumn,&errmsg) != SQLITE\_OK)

{

printf("user message search over.\n");

}

if(nrow >= 1)

{

/\*从员工信息表中删除数据\*/

sprintf(sql,"delete from usr\_data where id=%d;",usrMsg->id);

if(sqlite3\_exec(db,sql,NULL,NULL,&errmsg) != SQLITE\_OK)

{

usrMsg->flags[0] = DELE\_SUC;

send(newfd,usrMsg,sizeof(MSG),0);

printf("delete msg successfully!\n");

}

}

}

8. 查看员工信息处理函数

接收结构体中的信息。根据员工ID，获取 员工信息表 中相应的数据，通过strcpy函数拷贝内容给结构体。将结构体反馈给客户端。

void emp\_cat(int newfd,MSG \*usrMsg)

{

recv(newfd,usrMsg,sizeof(MSG),0);

sprintf(sql,"select \* from emp\_data where id=%d;",usrMsg->id);

if(sqlite3\_get\_table(db,sql,&resultp,&nrow,&ncloumn,&errmsg) != SQLITE\_OK)

{}

if(nrow >= 1)

{

printf("===== get employee message success.=====\n");

usrMsg->id = atoi(resultp[6]);

strcpy(usrMsg->name,resultp[7]);

usrMsg->age = atoi(resultp[8]);

strcpy(usrMsg->sex,resultp[9]);

strcpy(usrMsg->phone,resultp[10]);

strcpy(usrMsg->addr,resultp[11]);

send(newfd,usrMsg,sizeof(MSG),0);

}

}

9. 更新员工信息处理函数

接收结构体中的信息。根据员工ID更新 员工信息表 中相应的数据。将更新结果反馈给客户端。

void emp\_data\_update(int newfd,MSG \*usrMsg)

{

recv(newfd,usrMsg,sizeof(MSG),0);

sprintf(sql,"update emp\_data set age=%d,phone='%s',addr='%s' where id=%d;",\

usrMsg->age,usrMsg->phone,usrMsg->addr,usrMsg->id);

if(sqlite3\_exec(db,sql,NULL,NULL,&errmsg) != SQLITE\_OK)

{

printf("===== data update successfully!=====\n");

usrMsg->flags[0] = CGE\_SUC;

ret = send(newfd,usrMsg,sizeof(MSG),0);

}

}

10. 修改密码处理函数

接收结构体中的信息。根据员工ID更新 员工信息表 中的密码信息。将修改结果反馈给客户端。

void passwd\_change(int newfd,MSG \*usrMsg)

{

recv(newfd,usrMsg,sizeof(MSG),0);

sprintf(sql,"update usr\_data set passwd='%s' where id=%d;",\

usrMsg->passwd,usrMsg->id);

if(sqlite3\_exec(db,sql,NULL,NULL,&errmsg) != SQLITE\_OK)

{

printf("===== password change success.=====\n");

usrMsg->flags[0] = CGE\_SUC;

send(newfd,usrMsg,sizeof(MSG),0);

}

}

四、客户端功能介绍

1、主函数和程序流程

程序通过命令行传参的方式，初始化套接字并连接服务器，当连接成功就进入主函数，显示注册、登录、忘记密码和退出功能。

当用户输入字符’1’时，客户端进入注册函数，并向服务器发送“注册”信号，用户需要输入用户名、密码、姓名、性别、电话、年龄和地址等信息，当输入完成时，客户端发送信息结构体给服务器，服务器将数据写表，并返回“注册成功”信号，客户端进行信号判断并打印相应信息。

当用户输入字符’2’时，客户端进入登录函数，并向服务器发送“登录”信号，用户需要输入用户名和密码，客户端发送信息结构体，服务器进行读表判断用户名和密码是否成功，并向客户端发送一个“登录成功”信号和用户权限，客户端进行信号判断，如果登录成功，就打印相关信息，并进入权限判断，如果权限为用户，就进入用户的二级菜单，如果权限为管理员，就进入管理员的二级菜单。

当用户输入字符’3’时，客户端进入忘记密码函数，并向服务器发送“忘记密码”信号，用户需要输入工号和用户名来验证，客户端发送信息结构体，服务器进行读表，当工号和用户名是正确的返回“查找成功”信号，客户端进行信号判断，当是“查找成功”，用户输入新密码，客户端发送信息结构体，服务器进行写表，并返回“修改成功”信号，客户端进行信息判断并打印相关信息。

当用户输入字符’3’时，客户端向服务器发送“退出”信号，客户端关闭套接字并退出。

2、用户函数和程序流程

当权限为用户时，进入用户函数，功能有查询个人信息，修改个人信息，修改密码和退出功能。

当用户输入字符’1’时，客户端进入查询个人信息函数，并向服务器发送“查询”信号，用户输入自己的工号后，发送信息结构体，服务器进行查表，并返回查询成功信号和信息结构体，客户端打印姓名、年龄、手机号和地址。

当用户输入字符’2’时，客户端进入修改个人信息函数，并向服务器发送“修改”信号，用户输入自己的工号和修改的年龄、电话、地址后，发送信息结构体，服务器进行写表，并返回修改成功信号，客户端进行信号判断并打印相关信息。

当用户输入字符’3’时，客户端进入修改密码函数，并向服务器发送“修改密码”信号，用户输入自己的工号和修改的密码后，发送信息结构体，服务器进行写表，并返回修改成功信号，客户端进行信号判断并打印相关信息。

当用户输入字符’4’时，客户端发送“退出”信号，并返回上一级目录。

3、管理员函数和程序流程

当权限为管理员时，进入管理员函数，功能相比用户函数增加了添加用户和删除用户功能。

当用户输入字符’1’时，客户端进入删除用户函数，并向服务器发送“修改密码”信号，管理员输入工号，发送信息结构体，服务器进行写表，并返回删除成功信号，客户端进行信号判断并打印相关信息。

当用户输入字符’2’时，客户端进入增加用户函数，并向服务器发送“注册”信号，因为此处功能和注册函数相同，所以调用了注册函数。