**课 程 设 计 报 告**

**课程设计名称：**移动平台应用开发

**课程设计题目**：电话簿

院（系）：计算机与信息工程学院

专 业：软件工程

班 级：14网编

学 号：20141105070

姓 名：孙宇

指导教师：朝力萌

完成日期：2016.7.11

**目 录**

[第1章 概要设计 2](#_Toc406624046)

[1.1题目的内容与要求 2](#_Toc406624047)

[1.2总体结构 2](#_Toc406624048)

[第2章 详细设计 2](#_Toc406624049)

[2.1主模块 2](#_Toc406624050)

[第3章 调试分析 3](#_Toc406624051)

[第4章 使用说明与执行结果 4](#_Toc406624052)

# 第1章 概要设计

## 1.1题目的内容与要求

**内容：编写出一个简单计分器**

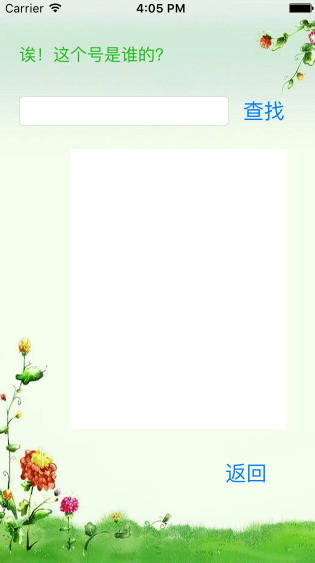
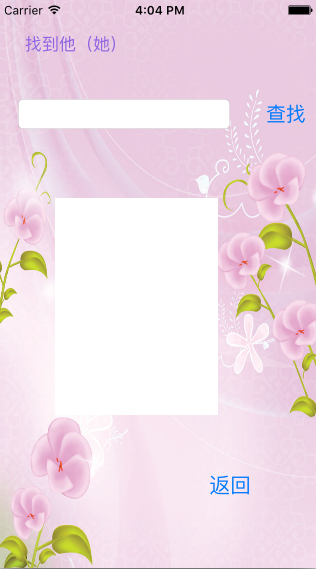
**要求： 利用swift语言编写出一个简单的计分器**

**可以实现倒计时、计分、查询历史纪录、录入队员信息等功能的计分器**

# 第2章 详细设计

## 2.1主模块

**运行程序界面：**

**  **

**程序主要代码：**

创建数据库：

添加联系人：

@IBAction func a(sender: AnyObject) {／／添加

let uname = self.name.text!

let uphone = self.phone.text!

let umail = self.mail.text!

let uadress = self.adress.text!

let sql = "insert into message(name,phone,mail,adress) values('\(uname)','\(uphone)','\(umail)','\(uadress)')"

print("aql:\(sql)")

let result = db.execute(sql)

print(result)

name.text=""

phone.text=""

mail.text=""

adress.text=""

}

显示全部联系人：

func initUser() {／／全部显示

let data = db.query("select \* from message")

for i in 0 ..< data.count {

var user=data[i]

show.text!+="用户名：" + String(user["name"]!) + "电话号：" + String(user["phone"]!) + "\n"

show.text!+="邮箱：" + String(user["mail"]!) + "地址:"+String(user["adress"]!) + "\n"

}

}

删除联系人（包括按电话删除和按名字删除）：

@IBAction func delet(sender: AnyObject) {／／删除

let Dname=a.text!

let Dphone=b.text!

if Dname==""{

if Dphone==""{

}else{

let sql="delete from message where phone = '\(Dphone)'"

print("aql:\(sql)")

let result=db.execute(sql)

print(result)

}

}else{

let sql = "delete from message where name = '\(Dname)'"

print("aql:\(sql)")

let result = db.execute(sql)

print(result)

}

a.text=""

b.text=""

}

@IBAction func deletall(sender: AnyObject) {//删除全部

let sql="delete from message"

print("aql:\(sql)")

let result = db.execute(sql)

print(result)

}

按姓名查找：

@IBAction func seek(sender: AnyObject) {／／姓名查找

var lookn = seekT.text!

let data = db.query("select \* from message where name='\(lookn)'")

for i in 0 ..< data.count {

var user=data[i]

seekup.text!+="用户名：" + String(user["name"]!) + "\n\n" + "电话号：" + String(user["phone"]!) + "\n\n"

seekup.text!+="邮箱：" + String(user["mail"]!) + "\n\n" + "地址:"+String(user["adress"]!) + "\n\n"

}

seekT.text=""

}

按电话查找：

@IBAction func seekphone(sender: AnyObject) {／／电话查找

var lookp = lookphone.text!

let data = db.query("select \* from message where phone='\(lookp)'")

for i in 0 ..< data.count {

var user=data[i]

textp.text!+="用户名：" + String(user["name"]!) + "\n\n" + "电话号：" + String(user["phone"]!) + "\n\n"

textp.text!+="邮箱：" + String(user["mail"]!) + "\n\n" + "地址:"+String(user["adress"]!) + "\n\n"

}

lookphone.text=""

}

修改联系人：

@IBAction func revise(sender: AnyObject) {／／修改

let name=rname.text!

let phone=rphone.text!

let mail=rmail.text!

let adress=radress.text!

if phone != ""{

let sql="update message set phone='\(phone)' where name='\(name)'"

print("aql:\(sql)")

let result = db.execute(sql)

print(result)

}

if mail != ""{

let sql="update message set mail='\(mail)' where name='\(name)'"

print("aql:\(sql)")

let result = db.execute(sql)

print(result)

}

if adress != ""{

let sql="update message set adress='\(adress)' where name='\(name)'"

print("aql:\(sql)")

let result = db.execute(sql)

print(result)

}

rname.text=""

rphone.text=""

rmail.text=""

radress.text=""

}

**数据库表的属性**

**数据库所建表的属性 name,phone,mail,adress**

# 第3章 调试分析

**对程序分别进行以下操作：**

1. **分别单击主界面四个按钮可以实现跳转**
2. **在计时中可以正确获取文本框中的内容，并且可以实现计时暂停等功能**
3. **计分可以实现基本加分功能，当意外退出可以实现继续使用，并可以存入相对应数据库表**
4. **参赛人员可以实现添加数据库，清空功能**
5. **历史纪录可以实现查询四个表中的数据，并且可以清空历史纪录并弹出对话框**

# 第4章 使用说明与执行结果

运行操作及结果:

通过调试分析可以实现连接数据库，并实现增删改查，基本实现计时器功能

**二．课程设计小结**

**在实验中发现——分析——总结的问题有：**

实现了电话簿的基本功能但是还有很多功能没有实现如不能实现：

1. 无法跨界面运用
2. 界面的转换还是有些生硬
3. 一些功能虽然初步实现，但并不完善有bug